

03063



UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO

UNIDAD ACADÉMICA DE LOS CICLOS PROFESIONALES  
Y DE POSGRADO DEL CCH

ADQUISICION DEL CONOCIMIENTO

**T E S I S**

QUE PARA OBTENER EL GRADO DE:  
**MAESTRO EN CIENCIAS  
DE LA COMPUTACION  
P R E S E N T A :**  
**SANDRA V SANCHEZ RIVERA**

ASESOR: DR. SERGIO MARCELLIN JACQUES

MEXICO, D. F.

1993

**TESIS CON  
FALLA DE ORIGEN**



## **UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso**

### **DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

ADQUISICIÓN  
DEL  
CONOCIMIENTO

## ADQUISICIÓN DEL CONOCIMIENTO

### INDICE

	PAGINA
INTRODUCCION	1
I. ADQUISICION DEL CONOCIMIENTO.	9
- Definición.	
- Antecedentes y Conceptos.	
- Aprendizaje.	
- Estados de la adquisición de conocimientos.	
- Fuentes principales de Adquisición del Conocimiento.	
II. EL INGENIERO DEL CONOCIMIENTO.	22
- Elección del Ing. del Conocimiento.	
- Proceso de interacción entre el experto y el Ing. del conocimiento.	
- Situaciones de conflicto durante la interacción con el experto del dominio.	
III. TECNICAS DE ADQUISICION DEL CONOCIMIENTO: ENFOQUE MANUAL.	33
- Entrevistas.	
* La entrevista como medio de Adquisición del conocimiento.	
* Participantes.	
* Planificación.	
* Transferencia de información.	
* Dialogo.	
- Análisis de Protocolo.	
- Adquisición del conocimiento de múltiples expertos.	
- Malla de Adquisición del Conocimiento.	
* Estructura de la malla de Adquisición del Conocimiento.	
* Estrategias para el uso de la Malla de Adquisición del Conocimiento	
- Casos o Ejemplos.	
- Inducción.	
- Técnica de la Malla de Repertorio.	

**IV. ADQUISICION DE CONOCIMIENTOS: ENFOQUE AUTOMATIZADO. 81**

- Adquisición del Conocimiento a través de técnicas de aprendizaje de conceptos.
  - \* Aprendizaje de conceptos.
  - \* Representación de conceptos y ejemplos.
  - \* Tendencias.
  - \* Interacción con el maestro.
  - \* Uso del marco de trabajo.
- Adquisición del conocimiento procedural de los Expertos del dominio.

**V. APLICACION. 105**

- Descripción del Sistema Experto.
- Adquisición del Conocimiento.
- Uso de la técnica de Malla de Repertorio.
- Conclusiones.

**BIBLIOGRAFIA Y REFERENCIAS. 125**

**APENDICE A. HERRAMIENTAS DE SOFTWARE PARA EL APOYO DE LA ADQUISICION DEL CONOCIMIENTO. 128**

- SIS. Un metasistema de entrevistas.
- SIS-More.
- IIS-LD.
- IIS-DB.
- El proyecto ESPRIT.

**APENDICE B. PROGRAMACION DE UN PROTOTIPO DE UN SISTEMA EXPERTO DE COMPONENTES REHUSABLES DE TIPOS DE DATOS ABSTRACTOS. 141**

- Introducción.
- Programas.

## INTRODUCCION

Como es de conocimiento general, hoy en día las computadoras son utilizadas en muchas áreas de aplicación. Su desarrollo ha sido tan rápido y amplio, en el sentido de que se ha ramificado en muchas y diferentes especializaciones, que cada área debe ser estudiada por expertos y especialistas diferentes.

Existe un área de investigación dentro de las Ciencias de la Computación que en los últimos años ha atraído mucho la atención de científicos e investigadores en el campo, esta área es la llamada "Inteligencia Artificial".

El término "Inteligencia Artificial" fue utilizado por primera vez por John McCarthy en una reunión en el Dartmouth College en 1956. En esta reunión estuvieron presentes varios pioneros como Marvin Minsky, Allen Newell, Herbert Simon y otros que han jugado roles muy importantes en la evolución de los conceptos de la Inteligencia Artificial.

La Inteligencia Artificial es el área dentro de las Ciencias de Computación que estudia la manera en la cual las computadoras pueden emular la inteligencia humana. Durante los primeros años de estudio de esta área, los investigadores se limitaban a desarrollar programas de juego, como el ajedrez, o programas que resolvieran rompecabezas o probaran teoremas de álgebra. Pero a medida que transcurrieron los años y las investigaciones y el interés fueron creciendo, se comenzó a incursionar en una subarea que involucraba el aprendizaje de la máquina, es decir la adquisición de nuevos conocimientos así como su aplicación y el desarrollo de experiencias con la consecuente modificación de su comportamiento.

Como resultado de estas investigaciones, inicialmente nació el interés en crear sistemas que resolvieran problemas universales, pero pronto se dieron cuenta que mientras más universales eran estos, menos satisfactorios eran los resultados obtenidos para cualquier tipo de problema. Como consecuencia de esto, los esfuerzos se enfocaron hacia la construcción de sistemas que resolvieran problemas específicos naciendo así la tecnología de los Sistemas Expertos.

El término "Sistema Experto" actualmente se refiere a un sistema que utiliza la tecnología computacional para almacenar e interpretar el conocimiento y la experiencia de un experto humano (en ocasiones varios expertos) en un área específica. Al acceder este conocimiento, un individuo debe poder obtener el beneficio de un "consejo experto" acerca de un área particular.

Hoy en día existe un gran número de aplicaciones diversas que utilizan la tecnología de los Sistemas Expertos, estas aplicaciones varían desde jugadores de ajedrez hasta aplicaciones para la explotación de minerales.

Con el propósito de establecer una idea más concreta de lo que expresa el término "Sistema Experto" se presentan de manera general sus componentes y la función que desempeñan dentro del sistema (Fig I-1):

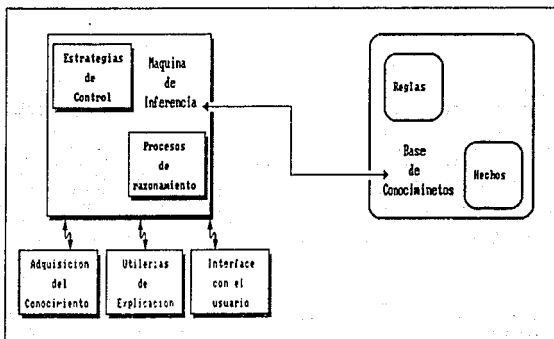


Figura I-1. Estructura de un Sistema Experto

- **La Base de Conocimientos.** La cual contiene el conocimiento específico sobre el área en la cual el sistema es experto. Consta de un conjunto de hechos y reglas que sirven para relacionar estos hechos e inferir otros. La Base de Conocimientos es un componente muy importante del Sistema Experto.
- **La Máquina de Inferencia.** Este es otro de los componentes principales de un Sistema Experto con el cual se controla el sistema. En la máquina de inferencia se interpretan y ejecutan las reglas contenidas en la Base de Conocimientos.
- **Interface con el usuario.** La cual permite al usuario introducir en un lenguaje de comandos en forma natural, instrucciones y datos para que el Sistema Experto trabaje.
- **Interface con el experto.** Módulo que permite realizar la adquisición de conocimientos. Permite captar la información proporcionada por el experto, e introducirla en la base de conocimientos. Este módulo es delicado pues del proceso exitoso de este depende que la base de conocimientos contenga la información adecuada.

Estos componentes sugieren que el elemento de mayor importancia en un Sistema Experto es la información y su calidad. Más aún, es posible afirmar que la calidad de la información inferida por un Sistema Experto depende de la calidad de la información almacenada, lo cual sugiere que la parte principal en la producción de un Sistema Experto es la Adquisición del Conocimiento.

A pesar de la importancia de la actividad de la Adquisición del Conocimiento, es relativamente poco lo que se ha investigado y publicado sobre este tema en relación a los estudios existentes sobre las demás actividades involucradas en el desarrollo de los Sistemas Expertos. Por ello considero de gran importancia realizar un estudio con la finalidad de concretar su significado, identificar algunos métodos apropiados, establecer la importancia del Ing. del Conocimiento en esta actividad, así como establecer el estado en el que se encuentra el desarrollo de herramientas de software para el apoyo a la Adquisición del Conocimiento y describir algunas de estas.



Este trabajo es realizado con el propósito de proporcionar una guía y una idea clara de cómo enfrentar el problema de la Adquisición del Conocimiento para aquellas personas de decidan desarrollar un Sistema Experto.

Las técnicas de Adquisición del Conocimiento descritas en este documento se dividen en dos (Fig I-2):

TESIS					
ADQUISICION DEL CONOCIMIENTO					
EL INGENIERO DEL CONOCIMIENTO					
TECNICAS					
ENFOQUE NO AUTOMATIZADO				ENFOQUE AUTOMATIZADO	
Entrevista	A. C. de Múltiples Expertos	Casos o Ejemplos	Inducción	A. C. a través de técnicas de aprendizaje de conceptos	A. C. procedural de los Expertos del dominio
APOYOS DE ANALISIS					
Análisis de Protocolo	Malla de Repertorio	Malla de Adquisición de Conocimiento			

Las técnicas que proponen un enfoque manual, es decir la intervención del Ing. del Conocimiento como intermediario entre la Base de Conocimientos y el experto del dominio; estas técnicas son:

- "Entrevista", la cual se desarrolla con detalle por ser el método más común de Adquisición del Conocimiento.
- "Análisis de Protocolo",
- "Adquisición del Conocimiento de múltiples expertos",
- y la de "Malla de Adquisición del Conocimiento".

Estas últimas son propiamente técnicas de apoyo a la entrevista que proporcionan esquemas de interpretación y análisis para la organización del conocimiento adquirido por el Ing. del Conocimiento.

Adicionalmente la técnica de entrevista, se mencionan otras menos conocidas como son las de:

- "Casos o Ejemplos",
- "Inducción",
- "Técnica de malla de repertorio",

Las técnicas que proponen un enfoque automatizado de Adquisición del Conocimiento. Estas conciben al el Ing. del Conocimiento como un filtro inadecuado a través del cual pasa el conocimiento y que por ser una persona con poco conocimiento y experiencia en el dominio de la expertez, podría modificar la esencia del conocimiento adquirido. Por esta razón sugieren la creación de una herramienta de software mediante la cual el experto introducirá su conocimiento a la Base de Conocimientos. Las técnicas que se mencionan en este documento son las de:

- "Adquisición del Conocimiento procedural de los expertos del dominio"
- "Adquisición del Conocimiento a través de técnicas de aprendizaje de conceptos".

Finalmente se describen algunas herramientas de software ya existentes como son:

- El metasistema de entrevistas SIS y algunos de los sistemas de entrevistas generados con este, como son:

I2S,  
 MORE,  
 IIS-LD,  
 IIS-DB.

- Finalmente se da una reseña de las investigaciones que se están realizando en el proyecto ESPRIT (Francia).

Este documento está compuesto por 5 capítulos cuyo contenido es el siguiente:

El capítulo I presenta la Adquisición del Conocimiento dentro del contexto de la creación de los Sistemas Expertos. Aquí se introduce la definición de la Adquisición del Conocimiento y los estudio realizados en esta área. Este capítulo también introduce el concepto de aprendizaje así como los tipos existentes. Finalmente establece tres estados o etapas de la Adquisición del Conocimiento. El estado 1 de recolección de información donde que se establece las tareas que realizará el Sistema Experto, quién utilizará el sistema y quien proporcionará la información. El estado 2 de representación de la tarea, donde se establece con detalle que actividades serán realizadas por cada tarea y el estado 3 donde se propone la construcción de un prototipo inicial.

El capítulo II introduce el concepto de Ing. del Conocimiento. Se establece su importancia en la construcción de un Sistema Experto y para ello se describen algunas de las habilidades con las que es importante que cuente. Se establecen algunos puntos importantes durante su interacción con el experto de dominio, de manera que juntos puedan trabajar como un equipo. Finalmente se mencionan algunas dificultades que pueden surgir durante esta interacción y se propone algunas posibles soluciones a tales conflictos.

En el capítulo III se presentan ocho técnicas de Adquisición del Conocimiento: Enfoque manual.

- "Entrevista" que es la técnica más común, donde se explica como interactuar con el experto y con otros ingenieros del conocimiento durante una sesión, como planearla y como preparar a los participantes,
- "Análisis de protocolo" que se basa en las transcripciones de la entrevista con el experto,
- "Adquisición del Conocimiento de múltiples expertos" que identifica los medios adecuados para entrevistar y obtener conocimiento de varios expertos,
- "Malla de Adquisición del Conocimiento" que es un método manual que propone la descripción de un espacio bidimensional con cinco formas de conocimiento experto y seis tipos básicos de preguntas de entrevistas con el propósito de proveer al Ing. del Conocimiento con una herramienta para detectar lo más relevante de la expertez,

- "Casos o Ejemplos" que estructura la adquisición en torno a casos de manera que propone el desarrollo de una estructura jerárquica con el fin de utilizarla como una librería de preguntas,
- "Inducción como herramienta de Adquisición del Conocimiento" propone la selección de un conjunto de formación o entrenamiento a partir de los cuales se puedan inducir reglas,
- "Malla de repertorio" que propone la creación de un sistema de referencias entre los constructores y los elementos del problema de manera que sea refinada hasta que el problema o la tarea quede claramente definido.

El Capítulo IV describe dos técnicas de Adquisición del Conocimiento: Enfoque automatizado.

- "Adquisición del Conocimiento a través de técnicas de aprendizaje de conceptos". Este método es tratado detalladamente por la diferencia de su enfoque y su importancia. Este método afirma que el conocimiento debe ser representado en forma abstracta debido que de esta manera el conocimiento puede ser almacenado y manipulado en un modo más efectivo. Se incorpora la descripción del aprendizaje de conceptos, su significado, la representación de los conceptos y ejemplos, la importancia de las búsquedas durante el aprendizaje de conceptos, así como las tendencias o influencias para tales búsquedas. Finalmente se describe la dependencia crítica de la interacción con el maestro durante el aprendizaje.

- "Adquisición del conocimiento procedural de los expertos del dominio" que sugiere que una parte importante de la expertez está formada por el conocimiento procedural.

En el capítulo V se presenta la descripción de la aplicación que consiste de la Adquisición del Conocimiento de un Sistema Experto de Componentes rehusables de Tipos de Datos Abstractos (TDA). La Adquisición del Conocimiento se realizó utilizando la técnica de entrevista durante la interacción con los expertos y la de Malla de Repertorio durante la transcripción de las entrevistas. Este Sistema Experto deberá permitir al usuario almacenar las especificaciones, conceptos, aplicaciones, así como implementaciones de los componentes reutilizables de los TDA y deberá guiar al usuario en la selección de los componentes adecuados. La Adquisición del Conocimiento se enfoca a obtener de los expertos en TDA las preguntas que permitirán guiar a un usuario no experto hacia los componentes adecuados.

El Apéndice A presenta algunas herramientas de software para el apoyo de la Adquisición del Conocimiento. En este capítulo se describe a SIS, un metasistema de entrevistas. SIS es una herramienta con la cual se pueden crear sistemas de entrevistas para la construcción de Sistemas Expertos. Así mismo se describen algunos sistemas de entrevistas como SIS-More, IIS-LD, IIS-DB que fueron creados con la utilización de SIS. En este capítulo también se presenta el proyecto ESPRIT desarrollado en Francia y se menciona las herramientas creadas bajo este proyecto. Finalmente se presenta un diagrama de algunas de estas herramientas y del tipo de conocimiento involucrado en su utilización.

CAPITULO 1ADQUISICIÓN DEL CONOCIMIENTO.

En gran parte de la literatura que existe acerca de los Sistemas Expertos ([1],[2],[3],[10] por citar algunos) es muy común encontrar frases que expresan que la Adquisición del Conocimiento es uno de los mayores retos durante la construcción de un Sistema Experto, que es la parte más importante del desarrollo, o que del éxito de esta actividad depende que un Sistema Experto infiera resultados adecuados o no. A pesar de ello, únicamente en la literatura especializada sobre aprendizaje o adquisición del conocimiento (que es escasa), se encuentra información detallada sobre este tema.

La Adquisición del Conocimiento se puede entender como un proceso que incorpora dos actividades : 1) La obtención de la expertez o conocimiento del experto y su almacenamiento en la base de conocimientos y 2) la generación de conocimiento nuevo por parte del Sistema Experto a partir del conocimiento adquirido del experto ( a lo cual en este documento se le llamará aprendizaje de la máquina para ubicar el contexto que se está tratando).

En el presente documento se entenderá la Adquisición del Conocimiento como la primera de las dos actividades antes mencionadas.

I. Definición.

Para entender con más claridad los conceptos hasta aquí mencionados y los que se desarrollarán a continuación, es necesario e importante entender los siguientes términos:

- Experto del dominio.- Es la persona cuyo conocimiento y experiencia son utilizados para producir información sobre un área de interés específica y posteriormente almacenarla en el Sistema Experto.

- **Ingeniero del conocimiento.** - Es la persona que trabaja con el experto del dominio para adquirir y codificar la información. Por la importancia de sus funciones se le dedica el siguiente capítulo para detallar sus actividades.

El "Conocimiento" dentro de la literatura de la Inteligencia Artificial ha sido asociado con estructuras formales a partir de las cuales se pueden realizar inferencias (Gaines)[26]. Sin embargo, gran parte del conocimiento humano no está adecuadamente estructurado para realizar representaciones formales. Ruses en la publicación de su libro "Diccionario Filosófico" en 1972, define al conocimiento como "más que una opinión y menos que una verdad". Los métodos clásicos de procesamiento de datos dan por hecho que los datos pueden ser asumidos como "verdad" y basan en estos sus inferencias. En los sistemas basados en conocimientos, se asumen los datos como "menos que una verdad pero más que una opinión" dado que es una opinión de un experto. Las implicaciones de estas "medias verdades" (inconsistencia, contradicción, cambio) son críticas durante la Adquisición del Conocimiento.

La Adquisición del Conocimiento es un proceso mediante el cual el Ing. del Conocimiento extrae el conocimiento (los hechos, reglas, procedimientos, etc.) de una fuente (libros, revistas, publicaciones, etc. y principalmente del experto del dominio) y toda la información necesaria para la construcción de un Sistema Experto; la analiza, la incorpora a la Base de Conocimientos y la refina. La Adquisición del Conocimiento es usualmente asociada con el propósito de expandir las capacidades de un sistema o mejorar su ejecución en alguna tarea específica. Para obtener tal información, el experto de dominio y el Ing. del Conocimiento deben trabajar juntos, siendo la disposición y buena voluntad de ambos un elemento básico. Esta interacción por su importancia se trata con mayor detalle en el siguiente capítulo.

Como ya se había mencionado, muchas personas estudiosas del área consideran que es la fase más importante en el desarrollo de un Sistema Experto, aunque también la más difícil por la naturaleza del proceso mismo.

Gran parte de la dificultad en la Adquisición del Conocimiento radica en el hecho de que no es fácil para el experto describir como él ve los problemas. Puede suceder que este no pueda distinguir entre los hechos o creencias y los factores que en ese momento tienen influencia en su toma de decisiones. Gran parte de su expertez consiste en la manera en que él ve los problemas, es decir su percepción. Este es esencialmente un problema psicológico.

## II. Antecedentes y Conceptos.

El problema de la adquisición de la expertez de una persona experta ya se había tratado en la literatura de Psicología [10]. Hawkins (1983) analizó la naturaleza de la expertez y enfatizó en sus enormes limitaciones y dependencia con suposiciones críticas que por lo general están implícitas. Bainbridge (1979, 1986) hizo notar que no existe necesariamente una correlación entre el comportamiento mental y los reportes verbales de tal comportamiento y que muchos psicólogos afirmaban que los datos verbales muchas veces eran inútiles. Dixon (1981) realizó estudios mostrando que gran parte de las actividades humanas no son conscientes. Collins (1985) sugirió que es posible que parte del conocimiento no sea accesible a través del experto, no solo porque este no sea capaz de expresarlo, sino también por no estar consciente de su significado. Los psicólogos clínicos ven este problema como una defensa cognitiva que limita la comunicación interna, y han desarrollado técnicas de interacción verbal para identificar los procesos cognitivos fundamentales (Freud 1914, Kelly 1955, Rogers 1967). Estos antecedentes son utilizados para eludir las defensas cognitivas, incluyendo aquellas resultantes de la automatización del comportamiento experto. Hayes-Roth, Waterman & Lenat (1983) proporcionaron algunas guías para la transferencia del conocimiento de un experto a un Sistema Experto.

Antes de enfrentarse a técnicas o métodos para la interacción con el experto del dominio, es importante establecer que el conocimiento experto por sí mismo ocasiona algunos problemas [10]:

- Conocimiento Fortuito: Los resultados obtenidos pueden depender de las características de la situación, la cual no es controlada por el experto.
- Conocimiento Inconsciente: Existe la posibilidad de que un experto no sea capaz de transmitir su expertez por medio de la crítica del comportamiento o ejecución de otros por no ser capaz de evaluar el suyo propio.
- Conocimiento no expresable en lenguaje: Es posible que un experto no sea capaz de transmitir su expertez explícitamente al no poder expresarla.



- Conocimiento no entendible cuando es expresado en lenguaje: Un aprendiz puede no ser capaz de entender el lenguaje en el cual es expresada la expertez.
- Conocimiento no aplicable aún cuando se expresa en lenguaje: Un aprendiz puede no ser capaz de convertir la comprensión verbal de las bases de la expertez en una ejecución o comportamiento experto.
- Conocimiento expresado que puede ser incorrecto: Los expertos pueden expresar frases que no corresponden a su comportamiento actual, lo cual provoca una ejecución incorrecta.

La transferencia lingüística de la expertez es uno de los pilares de la Adquisición del Conocimiento. Esta transferencia del conocimiento puede ser tan extensa o limitada como el sistema lo requiera. A continuación se desarrollan algunos métodos alternativos para tal transferencia según el nivel de conocimiento que se desea transmitir:

- La expertez puede ser transmitida al manipular el aprendizaje del medio ambiente: Un profesor puede ser capaz de establecer condiciones efectivas para que un aprendiz adquiera la expertez sin la necesidad de entender la habilidad o que él mismo sea un experto en ella.
- La expertez puede ser transmitida por evaluación. Un profesor puede ser capaz de inducir expertez indicando los comportamientos correctos e incorrectos sin necesariamente entender la habilidad en detalle o él mismo ser un experto en ella.
- La expertez puede ser transmitida por ejemplos. Un experto puede ser capaz de transmitir una habilidad mostrando su propio comportamiento o ejecución sin necesariamente entender las bases de su expertez.

Es importante establecer a que nivel se desea o se puede obtener el conocimiento experto que se introducirá en la Base de Conocimientos, ya que de ello dependerá el alcance de los resultados que se obtendrá del Sistema Experto.

### III. Aprendizaje.

El aprendizaje, es un concepto de gran importancia ya que es la base del estudio de la Adquisición del Conocimiento. Por ello es indispensable entender su significado e identificar los tipos de aprendizaje que existen.

Existe un gran número de definiciones de aprendizaje, sin embargo la que se muestra a continuación da una idea clara de lo que se debe entender como aprendizaje en el contexto que se desarrolla en el presente documento.

"El aprendizaje es el proceso que comprende la obtención de nuevos conocimientos así como su aplicación y desarrollo de experiencias con la modificación del comportamiento." [17]

El aprendizaje de conocimiento nuevo se realiza a través de diferentes métodos, dependiendo del tipo de material a ser aprendido, la cantidad de conocimiento relevante que ya se posea y el medio ambiente en el cual tiene lugar el aprendizaje.

A continuación se presenta una clasificación de los tipos de aprendizaje. Esta clasificación sirve como una guía en el estudio o comparación de las diferencias entre estas. Esta clasificación es independiente del dominio del conocimiento y del esquema de representación utilizado. Esta basado en el tipo de estrategia de inferencia.

- 1.- Aprendizaje por memorización. Es la adquisición directa de nuevos conocimientos. Es la forma más simple de aprendizaje. Requiere la menor cantidad de inferencia. En un Sistema Experto se realiza mediante la copia del conocimiento en la misma forma que será utilizada directamente en la base de conocimientos.
- 2.- Aprendizaje por instrucción directa o algorítmica. Consiste en transformar la información desde un lenguaje fuente a una representación interna y posteriormente interpretarla en función de la utilización requerida. Se utiliza este tipo de aprendizaje cuando un profesor presenta directamente un número de hechos en una manera bien organizada.

- 3.- Aprendizaje por analogía. Es el proceso de aprendizaje de conceptos o soluciones nuevas a través del uso de conceptos o soluciones similares ya conocidas, con la consecuente adquisición de experiencia. Se utiliza este tipo de aprendizaje, por ejemplo cuando se reescribe un programa para ejecutar funciones similares.
- 4.- Aprendizaje por inducción. Es un tipo de aprendizaje utilizado frecuentemente. Es una forma poderosa de aprendizaje que necesita una gran cantidad de inferencia. Esta forma de aprendizaje requiere el uso de inferencia inductiva. Se utiliza el aprendizaje inductivo cuando se formula un concepto general después de haber revisado varias instancias o ejemplos de tal concepto. Por ejemplo se aprende el concepto de color o sabor dulce después de haber experimentado la sensación asociada con muchos objetos de color o comidas dulces.
- 5.- Aprendizaje por Deducción. Se realiza a través de una secuencia de pasos de inferencia deductiva utilizando hechos conocidos. A partir de los hechos conocidos, los nuevos hechos o relaciones son lógicamente derivados. El aprendizaje deductivo usualmente requiere mayor cantidad de inferencia que los otros métodos.

Adicionalmente a la clasificación presentada, en algunas ocasiones se hace referencia a métodos de aprendizaje como métodos débiles o métodos ricos de conocimiento. Los métodos débiles son métodos de propósito general en los que ningún o poco conocimiento inicial está disponible, algunos ejemplos de este tipo de métodos son los Algoritmos Genéticos y los Automatas de aprendizaje. Estos métodos son más mecánicos que los clásicos métodos ricos de conocimiento. Estos últimos, a menudo cuentan con una forma de búsqueda heurística en el proceso de aprendizaje, un ejemplo común de este tipo de métodos es el aprendizaje inductivo.

#### IV. Estados de la Adquisición del Conocimiento.

El proceso de adquisición del conocimiento no consiste de un número preciso de pasos, sin embargo se pueden identificar tres de ellos que son los siguientes[3]:

- Estado 1. Recolección de información.
- Estado 2. Representación de tareas.
- Estado 3. Construcción de un prototipo inicial.

La adquisición del conocimiento es un proceso cíclico como se muestra en la Fig 1-1 que comienza con la elección del Ingeniero del Conocimiento y culmina cuando este y el o los expertos del dominio formulan y representan la tarea. El proceso comienza a disminuir a medida que el prototipo está más cerca de cubrir todos los requerimientos de la tarea.

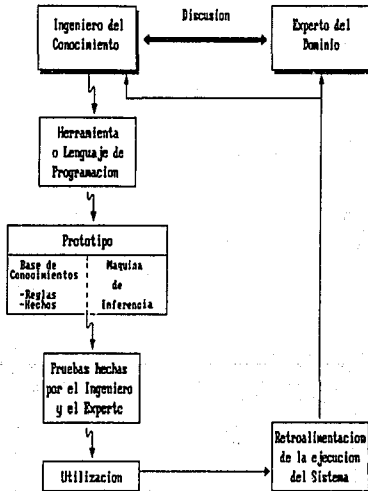


Fig 1-1. Ciclo de la Adquisición del Conocimiento [2]

#### Estado 1. Recolección de información.

- Algunas veces se involucran a muchos expertos en la construcción de un sistema. Sin embargo por lo general existe una interacción entre un experto y un ingeniero del conocimiento. Juntos definen los elementos claves de la tarea. El experto puede limitar el alcance de la tarea definida, o en algunas ocasiones las decisiones o tareas pueden ir más allá del alcance del sistema.

El experto del dominio y el ingeniero del conocimiento, deben definir que incluir y que excluir del sistema. Posteriormente estos deben comunicar estas definiciones de las tareas a otros individuos involucrados en el proceso, especialmente a los directivos. Muy a menudo sucede que los directivos se decepcionan pues el tamaño del Sistema Experto no es el que esperaban. Esta dificultad radica fundamentalmente en la falta de comunicación durante el desarrollo de los primeros prototipos.

Posteriormente los desarrolladores comienzan a caracterizar y definir la tarea, la mayoría de las tareas tienen una estructura. A un nivel elemental, el experto define el alcance y naturaleza de las tareas al ingeniero, el cual debe determinar que subtareas están involucradas y si son segmentables.

Una vez que el ingeniero del conocimiento tiene una definición trabajada de la tarea, determina los hechos que el experto utiliza en su realización. Además de los hechos, el ingeniero del conocimiento aprende la terminología que el experto utiliza, posteriormente programa las tareas, definidas en lenguaje informal, en reglas del sistema.

A su vez es de gran importancia que el ingeniero conozca los requerimientos finales del usuario, y es esencial que conozca los factores que podrían en un momento dado obstruir un desempeño exitoso. Como ya se había mencionado, la certidumbre de las salidas del sistema, depende de la certidumbre de la información de entrada.

En algunas situaciones donde no se tiene certeza sobre la veracidad de la información, el experto emplea bastante tiempo, realizando una doble revisión de la información, y los resultados pueden ser cuestionables. En tal caso se debe determinar si esto afecta en la productividad del sistema.

Mientras se realiza la recolección de la información, el ingeniero del conocimiento identifica los conceptos claves, los hechos y sus relaciones. En el siguiente estado el ingeniero debe seguir probando para determinar como el experto utiliza estos conceptos, reglas y relaciones.

Al finalizar este estado, el Ing. del Conocimiento debe haber establecido los siguientes puntos:

- Cuáles son las tareas específicas que realizará el Sistema Experto.
- Quién utilizará el sistema.
- Cómo se involucrará al sistema en el medio ambiente actual.
- Quién proporcionará el conocimiento y la expertez.

- Estado 2. Representación de tareas.

En este estado el ingeniero debe conocer casi todos los hechos que utiliza el experto. Ahora su tarea consiste en aprender el proceso que sigue el experto para realizar la tarea. Dado que el experto puede emplear "reglas de dedo", presentimientos o intuiciones, el ingeniero debe definir estos procesos y determinar por que los usa el experto.

Esto puede ser representado en un diagrama que muestre la secuencia que deben seguir los hechos para facilitar la tarea del ingeniero.

Para obtener una comprensión minuciosa de los procesos del experto, el ingeniero del conocimiento debe ser capaz de responder detalladamente las tres siguientes preguntas:

1.- *Qué pasos toma el experto en la realización de una tarea?* El conocimiento estratégico del experto indica el proceso mediante el cual se realiza la tarea. Cualquier tarea puede ser vista como una secuencia de pasos. El experto puede realizar una serie fija de pasos o puede seleccionar diferentes pasos en diferentes casos. Algunas tareas solo pueden ser completadas si algunos pasos específicos son ejecutados en un orden especial. Para otras el orden es irrelevante.

2.- *Cómo razona el experto a partir de los datos de entrada hacia los datos de salida?* El conocimiento

de juicio del experto especifica cómo el experto utiliza la información que es conocida sobre el caso para inferir información adicional acerca del caso.

3.- *Qué características del caso utiliza el experto en la ejecución de la tarea?* Las definiciones de las características relevantes del caso constituyen el conocimiento real del experto. Las características mismas incluyen todos los hechos e hipótesis acerca del caso que el experto utiliza y deduce.

Las respuestas a estas tres preguntas proveen la información que el ingeniero del conocimiento necesita para comenzar la implementación del Sistema Experto.

Finalmente el ingeniero debe descubrir cualquier contratiempo que pueda afectar este proceso. Sin embargo este tipo de contratiempos no son los únicos a considerar. Un contratiempo son los errores que pueden existir en la representación. Estos ocurren cuando existe una diferencia entre la manera en la que el experto establece sus hechos, reglas y procedimientos, y la manera en la que los representa el sistema.

Las herramientas para la construcción de sistemas que utilizan interfaces en lenguaje natural, suavizan el problema pues permiten al experto interactuar directamente con el sistema. Esta interacción requiere el uso de una herramienta adecuada por lo que el ingeniero debe conocer las capacidades y limitaciones de la herramienta. Por ejemplo si el ingeniero del conocimiento sabe que el experto posee una forma particular de resolver un problema, y que esta refleja encadenamiento hacia adelante, sería un error seleccionar una herramienta que sólo haga encadenamientos hacia atrás.

#### - Estado 3. Construcción de un prototipo inicial.

Una vez que el ingeniero entiende los hechos, reglas y procedimientos, da comienzo al proceso de documentación. Si alguno de los hechos tiene

incertidumbre asociada, el ingeniero debe resolverla durante el estado de documentación.

Si la interpretación de los hechos depende de un orden particular de procesamiento, el ingeniero debe saberlo.

Después de esto, los desarrolladores deben trazar los hechos, reglas y procedimientos necesarios para realizar la tarea y ponerlos en un esquema de representación documentado. Estos esquemas reflejan las capacidades de la herramienta de construcción elegida.

Para realizar la tarea, el experto debe realizar preguntas específicas en una secuencia apropiada de fuentes particulares, y el ingeniero debe saber como respondería el experto. Adicionalmente, el ingeniero debe saber que preguntas debe hacer el sistema para obtener la información adecuada.

Estas preguntas también deben ser documentadas.

#### V. Fuentes principales de Adquisición del Conocimiento.

Las características fundamentales de los componentes de la Adquisición del Conocimiento son mas claramente analizados si se considera la naturaleza de los paradigmas de los sistemas clásicos y de los sistemas expertos. Es obvio que los Sistemas Expertos utilizan algunas técnicas computacionales nuevas, pero la diferencia importante respecto a los sistemas clásicos radica en los cambios fundamentales que existen en su diseño. El enfoque clásico en el diseño de sistemas de diseño y control involucra la instrumentación, recolección de datos, modelación y secuencias de optimización como se ve en la Fig 1-2:

- El conocimiento de casos de historias pasadas es utilizado para seleccionar una clase de los modelos del sistema.
- La información requerida para discriminar entre estas clases determina la manera en la que el sistema deberá ser instrumentado para la Adquisición de Datos.



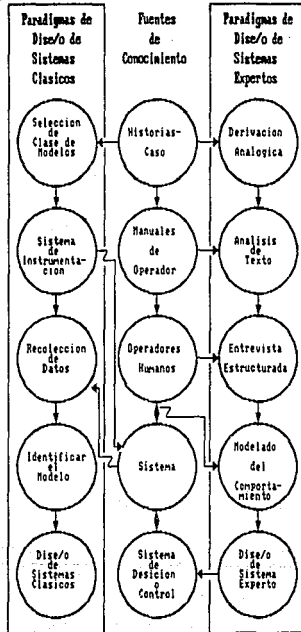


Fig. 1-2 Paradigmas de diseño de Sistemas Clásicos y Sistemas Expertos

- Los datos para el sistema son recolectados a través de la instrumentación.
- A partir de el grupo de modelos, se identifica el que mejor modele a los datos.
- Este modelo es utilizado para diseñar un sistema de decisión o control con la finalidad de obtener una ejecución óptima.

Este enfoque es exitoso únicamente si los sistemas bajo consideración pueden ser instrumentados y modelados. Su éxito radica en los casos en los que el sistema a ser controlado es por sí mismo un artefacto humano.

El paradigma de los Sistemas basados en conocimientos es particularmente aplicable cuando no es posible modelar el sistema pero existe una fuente alternativa de datos dado que los operadores humanos están capacitados para ejecutar una tarea de decisión o control. La columna derecha de la Fig 1-2 muestra el uso de las fuentes de conocimiento dentro del paradigma del diseño de los Sistemas Expertos:

- Una entrevista estructurada puede ser utilizada para adquirir conocimiento directamente de los operadores.
- El modelado del comportamiento puede ser utilizado para identificar las estrategias del operador aun si el no esta consiente de ellas o proporciona estrategias incorrectas durante las entrevistas.
- El análisis de textos puede ser utilizado con material de instrucción como manuales del operador.
- El razonamiento por analogía puede ser utilizado en los casos-historia sin la necesidad de realizar recolección de datos típicos de los diseños de sistemas clásicos.

Estos paradigmas de diseño de sistemas, así como las diferentes técnicas de Adquisición del Conocimiento para Sistemas Expertos no necesariamente deben ser competitivas. En su lugar pueden ser utilizadas en la Adquisición del Conocimiento durante el diseño de Sistemas Expertos.

## CAPITULO 2

### EL INGENIERO DEL CONOCIMIENTO.

El término Ingeniería del Conocimiento es usado para describir el proceso de la construcción de Sistemas Expertos, y por lo tanto quienes los construyen se conocen como ingenieros del conocimiento.

El ingeniero del conocimiento desempeña un papel fundamental en el desarrollo del sistema. Otras personas pueden proporcionar la información de entrada, pero es el ingeniero del conocimiento quien debe analizarla y estructurarla.

Un Ingeniero del Conocimiento ideal debe estar familiarizado con muchas áreas como psicología computacional, lógica formal, estadística, así como con la terminología utilizada por el experto del dominio.

El modelo básico de la Ingeniería del conocimiento está sustentado en el hecho de que el Ing. del Conocimiento debe actuar como un mediador entre el experto y la Base de Conocimientos, extrayendo el conocimiento del experto, codificandolo en la Base de Conocimientos y refinandolo en colaboración con el experto hasta alcanzar una ejecución aceptable. La Fig 2-1 muestra este modelo básico de adquisición manual del conocimiento a partir de un experto:

- El Ing. del Conocimiento entrevista al experto para extraer su conocimiento.
- El Ing. del Conocimiento codifica el conocimiento adquirido en la Base de Conocimientos.
- El shell utiliza la Base de Conocimientos para realizar inferencias sobre un caso en particular.
- Los usuarios finales utilizan el shell para obtener "consejo" sobre casos particulares.

A continuación se describen las actividades de la gente que trabaja con el ingeniero del conocimiento.

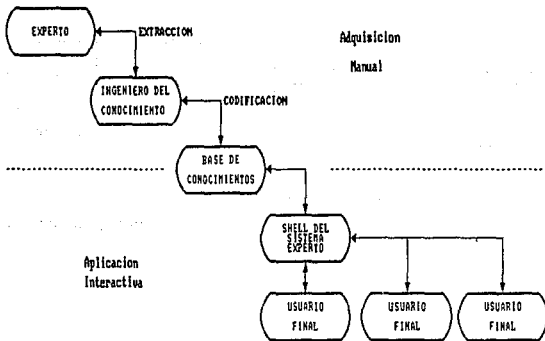


Fig 2-1 Modelo básico de Ingeniería del Conocimiento

### La Dirección.

Los directivos deben inicialmente definir y aprobar la tarea a desarrollarse. Posteriormente seleccionan un equipo apropiado y son los que finalmente eligen al ingeniero del conocimiento y a el (los) experto(s) del dominio asignados al proyecto. El soporte proporcionado por los directivos es básico para el desarrollo de un Sistema Experto útil, pues son quienes proporcionan los recursos, equipo y tiempo.

### Los Expertos.

Para construir un Sistema Experto útil se requiere que exista una interacción intensiva entre el ingeniero del conocimiento y el experto del dominio. Este tema se tratará con mayor detalle en la sección II.

### Los Usuarios Finales.

Los usuarios finales son probablemente la fuente más importante de información y dado que son los que utilizarán el Sistema Experto; es deseable que estén involucrados desde el principio del desarrollo.

La información que proporcionan los usuarios involucra los siguientes tópicos:

- Las necesidades para la aplicación.
- La terminología utilizada.
- Niveles de habilidad.
- Los recursos disponibles para responder preguntas.
- Cuando la solución es total.

### I. Elección del Ing. del Conocimiento.

Para seleccionar a el Ing. del Conocimiento o al equipo de Ings. del Conocimiento, se deben tener en cuenta que es deseable que estos cuenten con ciertas habilidades. El Ing. del Conocimiento que sea integrado a un proyecto debe poseer fuertes habilidades en ingeniería de software, en ingeniería del conocimiento y habilidades interpersonales.

#### **Habilidades en Ingeniería de Software.**

Un Sistema Experto finalmente es un programa, por lo cual las habilidades de el o los Ing(s). del Conocimiento deben incorporar la comprensión de algoritmos fundamentales y estructuras de datos, así como habilidades para evaluar la eficiencia de algoritmos alternativos y representaciones de datos. La habilidad para el diseño del software es importante para el desarrollo de una Base de Conocimientos bien diseñada que pueda ser entendida y extendida por las personas que realicen su mantenimiento.

Dependiendo del medio ambiente de operación del Sistema Experto, el Ing. del Conocimiento y su equipo necesitarán habilidades de diseño de interfaces con el usuario y habilidades de integración de software.

**Habilidades en Ingeniería del Conocimiento.**

El Ing. del Conocimiento debe transformar el conocimiento del experto en una Base de Conocimientos. Para realizar esta transformación, este debe estar familiarizado con el shell del Sistema Experto que es utilizado para construir el sistema. Debe entender la inferencia incorporada y los mecanismos de control y razonamiento provistos por la máquina de inferencia.

El Ing. del Conocimiento debe saber qué formalismos de representación soporta el shell y cómo la máquina de inferencia utiliza cada forma de representación. Adicionalmente, si se cuenta con varios shell's, el Ing. del Conocimiento deberá ser capaz de evaluar los intercambios entre las maneras alternativas de representar el mismo conocimiento.

En caso de que el proyecto desarrollara su propio shell en lugar de utilizar uno ya existente, se necesitará una base sólida de conocimientos en Inteligencia Artificial. Finalmente es importante que el Ing. del Conocimiento este familiarizado con varios paradigmas de control, inferencia y representación.

**Habilidades Interpersonales.**

La Adquisición del Conocimiento depende fuertemente de las habilidades interpersonales del Ing. del Conocimiento, y de manera especial de su habilidad de comunicarse efectivamente y de trabajar adecuadamente con otros.

Para dirigir al experto en diversos campos con el fin de articular sus estrategias de resolución de problemas y razonamiento, el Ing. del Conocimiento debe expresarse de manera clara y precisa.

La naturaleza colaborativa del desarrollo de un Sistema Experto requiere que el Ing. del Conocimiento sepa trabajar en equipo. Debe ser capaz de trabajar cooperativamente con los otros miembros del proyecto, ser sensitivo tanto al talento como a las necesidades de sus compañeros de equipo.

**II. Proceso de Interacción entre el experto y el Ing. del Conocimiento.**

Esta interacción consiste de entrevistas realizadas en un periodo de tiempo. Un ingeniero experimentado, sabe que un experto asume que muchos hechos o procedimientos son evidentes, de manera que parte de su expertez puede no estar definida. Si esto ocurre, el sistema podría resultar debilitado o más aún producir una falla. Para minimizar este riesgo, el ingeniero debe hacer preguntas que ayuden al experto a definir los detalles. La Fig. 2-2 muestra un diagrama de algunos elementos de dicho proceso de interacción.

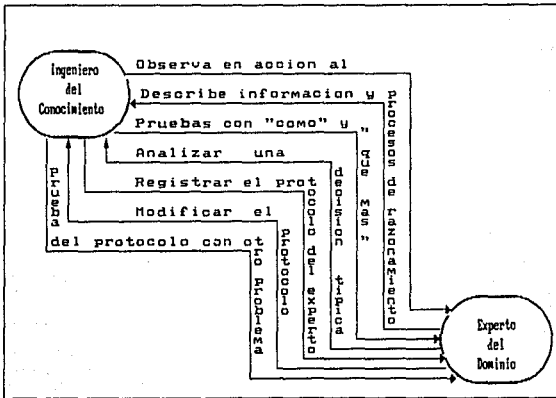


Figura 2-2. Proceso de Interacción

Inicialmente el ingeniero del conocimiento debe observar como el experto realiza las tareas y toma decisiones, ya que de esta manera el experto describe los tipos de información, conocimiento y procedimientos que incorporan sus actividades.

De esta manera el ingeniero adquiere la suficiente información para comenzar a escribir un protocolo general de los procesos del experto.

Este protocolo es un trazo de los pasos realizados por el experto para realizar una tarea. Este trazo sirve como base para cuestionar al experto. Si el ingeniero encuentra inconsistencias o piezas faltantes en el protocolo, debe realizar las preguntas apropiadas para obtener más información.

El propósito de la elaboración de este protocolo, es documentar lo que hace el experto en la realización de una tarea.

Una vez que el ingeniero ha entendido todo el proceso, la herramienta constructora del sistema puede ser seleccionada. Finalmente, cuando el protocolo es más específico, el ingeniero crea una versión de un prototipo inicial. El experto presenta al ingeniero una serie de situaciones para probar el prototipo en la toma de decisiones. Una vez que el prototipo pasa todas las pruebas, comienza la codificación.

En ocasiones los expertos suelen desperdiciar mucho tiempo enfocándose en situaciones excepcionales en lugar de enfocarse en las actividades y decisiones de rutina.

Después de analizar esta información preliminar, el ingeniero del conocimiento produce varios resultados, algunos de los cuales formarán parte de la documentación. Entre los resultados producidos está la documentación de los procedimientos que el experto sigue así como los hechos a tomarse en consideración. El proceso de análisis ayuda también a tomar una decisión sobre la selección de la herramienta para la construcción del sistema que se utilizará en el desarrollo de la base de conocimientos. La mejor herramienta será aquella que encaja con el esquema de representación de la tarea. Si la tarea incluye el descubrimiento de información nueva, una herramienta que no utilice encadenamiento hacia adelante, es inapropiada. Por lo tanto el ingeniero del conocimiento debe tener un conocimiento minucioso de la tarea para seleccionar la herramienta con la que él y el experto puedan trabajar juntos para producir prototipos que conduzcan a un sistema que cubra todos los requerimientos de la tarea.



### **Identificación del Experto del dominio.**

En proyectos simples, un solo experto tiene tanto el conocimiento como la disponibilidad de tiempo para proveer todos los requisitos. Este experto tiene la autoridad para juzgar cuando el comportamiento del sistema es correcto. En otros proyectos, aplicaciones más complicadas, es necesaria la inclusión de más de un experto del dominio debido a que no existe una sola persona que tenga conocimiento de todos los aspectos de la tarea.

Los expertos del proyecto pueden ser seleccionados en reuniones que involucren al Ing. del Conocimiento, a la gente que ha sido seleccionada como posibles expertos del proyecto y a los directivos de cada experto. El Ing. del Conocimiento y los expertos deben discutir la tarea propuesta para decidir quien o quienes pueden proveer la expertez necesaria para cada componente de la tarea. El Ing. del Conocimiento debe proporcionar los estimados en tiempo que serán necesarios de cada experto, para que posteriormente los expertos y sus directivos decidan si el tiempo y las responsabilidades actuales les permiten trabajar en el proyecto o no.

Una vez que los expertos han sido identificados, el Ing. del Conocimiento, los expertos y sus directivos deben decidir la manera en la que los expertos serán incluidos en el proyecto. Si el proyecto obtiene información de un solo experto, este será miembro del equipo. Si el proyecto incluye a más de un experto, el grupo debe decidir cuando el equipo debe incluir a un experto o a más de uno.

### **III. Situaciones de Conflicto durante la interacción con el experto.**

Los problemas que de manera general debe enfrentar el Ing. del Conocimiento surgen a partir de la interacción con el experto de dominio. A continuación se presentan los problemas más frecuentes acompañados con un planteamiento de posibles soluciones.

Durante el proceso de adquisición del conocimiento, pueden presentarse una serie de problemas originados cuando el experto intenta explicar lo que él hace. A menudo los expertos utilizan terminología característica de su área y hasta en ocasiones crean su propia terminología para describir hechos y procesos de su propia experiencia. Esta terminología por lo general no es conocida por personas fuera del campo.

Por lo antes mencionado la primera dificultad para el ingeniero del conocimiento surge del desconocimiento de la terminología del experto.

La segunda dificultad es que el experto tiende a omitir muchos detalles específicos necesarios para realizar la tarea, por lo que el ingeniero debe estar seguro de que la mayor parte de los detalles están contemplados y explicados tan detalladamente como sea posible.

Si se ignoran las dificultades que el experto desarrolla durante la interacción con el Ing. del Conocimiento, este se vuelve poco cooperativo o no disponible. También es posible que el experto llegue a cansarse de estar aparentemente repitiendo y explicando muchas veces lo que para él es obvio. Si no se le enseñan resultados rápidos y no se puede mantener su entusiasmo sobre el proyecto, este podría dejar de prestar la suficiente atención a los detalles y afectar directamente a los resultados.

Por lo antes mencionado es adecuado dejar saber al experto lo que se está haciendo con la información adquirida, enseñar al experto a dirigir el proceso, animarlo a volverse más independiente de los modelos de trabajo y del área del problema que él utiliza en la resolución de los problemas.

Por otra parte si se puede obtener un nuevo caso del experto e implementarlo en pocos días, habrá muchas oportunidades de obtener una retroalimentación de como se está haciendo el sistema. Si el tiempo invertido entre la articulación de nuevos conocimientos y enseñar este conocimiento en uso, es corto, se tienen muy buenas posibilidades de mantener motivado a todo el equipo involucrado, aunque por supuesto, existirán periodos en los que toda la base de conocimientos necesite ser reestructurada. Esto puede involucrar varias semanas durante las cuales el Ing. del Conocimiento se

enfoque en el diseño y codificación en lugar de la adquisición de nuevos conocimientos.

Debido a que es difícil tanto para el experto como para el Ing. del Conocimiento manipular todo el flujo de la conversación y probar los detalles del proceso de razonamiento del experto. Es mucho más fácil y menos estresante si dos Ings. del Conocimiento dividen estos roles [1]. Se pueden alternar los roles para mantenerlos frescos y al día.

### III. Utilización del herramientas automatizadas de Adquisición del Conocimiento.

Con el transcurso de los años y el avance de la tecnología, la labor del Ing. del Conocimiento como intermediario entre el experto y la Base de Conocimientos ha sido cuestionada no solo por los altos costos, sino también por su efectividad, es decir de la pérdida de conocimiento a través del intermediario.

Tales consideraciones condujeron a la posibilidad de realizar la Adquisición del Conocimiento mediante herramientas automatizadas con las cuales el experto interactúe directamente. La Fig 2-3 muestra como rol que desempeña el Ing. del Conocimiento ha ido cambiando a medida que se han ido utilizando tales herramientas.

La Adquisición del Conocimiento interactiva así como las herramientas de codificación pueden reducir bastante la necesidad de que el Ing. del Conocimiento actúe como intermediario, sin embargo en la mayoría de las aplicaciones existe un rol importante para el Ing. del Conocimiento. Como se ve en la Fig 2-3 el Ing. del Conocimiento es responsable de:

- Aconsejar al experto sobre el proceso interactivo de Ing. del Conocimiento.
- Administrar las herramientas interactivas de Adquisición del Conocimiento, configurandolas adecuadamente.
- Editar la Base de Conocimientos no codificada en colaboración con el (los) experto(s).
- Administrar las herramientas de codificación de conocimiento, configurandolas apropiadamente.

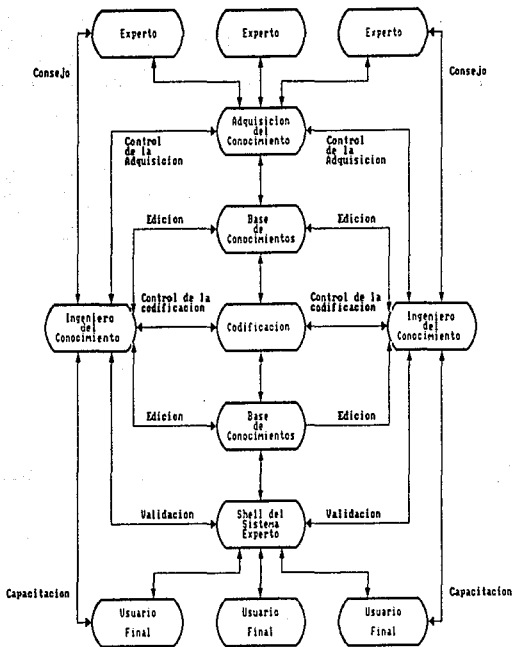


Fig 2-3. Rol del Ing. del Conocimiento utilizando herramientas automatizadas de Adquisición del Conocimiento.

- Editar la Base de Conocimientos codificada en colaboración con el (los) experto(s).
- Validar la aplicación de la Base de Conocimientos en colaboración con el (los) experto(s).
- Configurar la interface con el usuario en colaboración con el (los) expertos y usuarios finales.
- Capacitar a los usuarios finales para el uso efectivo de la Base de Conocimientos en colaboración con el (los) experto(s) desarrollando procedimientos operacionales y de capacitación.

La utilización de herramientas automatizadas para la Adquisición del Conocimiento puede ser combinada con la adquisición manual. El Ing. del Conocimiento puede directamente extraer el conocimiento del experto y utilizar las herramientas de adquisición interactiva para introducir el conocimiento en la Base de Conocimientos.

### CAPITULO 3

#### TÉCNICAS DE ADQUISICION DEL CONOCIMIENTO:

##### ENFOQUE MANUAL

La primera generación de herramientas de Adquisición del Conocimiento nació en los 70's y esta basada en la entrevista entre el Experto de dominio y el Ing. del Conocimiento. La segunda generación en los 80's ha sido basada en aplicaciones (auxiliadas por la computadora) de técnicas específicas como mallas de repertorio, modelado del comportamiento y análisis de texto. Actualmente (90's) las investigaciones de la tercera generación están enfocadas a las herramientas que proveen ambientes de Adquisición del Conocimiento altamente integrados que ofrecen una gran cantidad de herramientas y técnicas. En este capítulo se contemplarán algunas herramientas manuales que están basadas en la intermediación del Ing. del Conocimiento como único vínculo entre el experto y la Base de Conocimientos.

A pesar de la existencia de herramientas automatizadas de Adquisición del Conocimiento, existen fuertes razones para que se sigan realizando investigaciones sobre métodos manuales:

- (1). La adquisición manual del conocimiento es el método más ampliamente utilizado y el más rutinario, además de que los métodos manuales pueden ser aplicados a cualquier dominio.
- (2). A pesar de que las características de la Adquisición del Conocimiento automatizada son factibles para algunos aspectos del proceso, los métodos manuales pueden ser utilizados para complementar y cubrir el contenido no accesible a los métodos puramente automatizados.
- (3). Los métodos manuales de entrevistas son los únicos útiles al tomar en cuenta las considerables diferencias de experto a experto y de dominio a dominio.

- (4). La última razón para enfocar la atención en los métodos manuales de adquisición de la expertez es que son utilizados en un rango creciente y variedad de escenarios, además de que sugieren direcciones prometedoras de cómo la expertez puede ser transferida más efectivamente con métodos automatizados.

En la adquisición del conocimiento no existe una metodología reconocida o que se haya probado que es universalmente efectiva. Algunos métodos trabajan bien con ciertos dominios y mal con otros y aunque es fácil juzgar si un método será de poca utilidad, es difícil asegurar cual será el idóneo. Muchos de estos métodos no son originales en el sentido que no fueron creados como una necesidad para la adquisición del conocimiento para Sistemas Expertos y han sido utilizados en otras áreas de investigación.

A continuación presentan 7 técnicas manuales de Adquisición del Conocimiento:

- la Entrevista, que es la más generalizada,
- Análisis de Protocolo,
- Adquisición del conocimiento de múltiples expertos, y
- Malla de Adquisición del Conocimiento,

estas son técnicas de apoyo a la entrevista,

- Casos o Ejemplos,
- Inducción,
- Técnica de la Malla de Repertorio,

estas últimas son técnicas que proponen opciones independientes a la realización de una entrevista.

## I. ENTREVISTA.

El método obvio de adquisición de conocimiento es identificar a un experto y cuestionarlo o entrevistarle.

La entrevista es el método más común para encontrar hechos en el análisis de sistemas. Este trabajo es intensivo y analítico y requiere tanto de buenas habilidades interpersonales, como de habilidades analíticas. Esto mismo es lo que ocasiona el cuello de botella en la adquisición del conocimiento pues limita la rápida propagación de las aplicaciones de Sistemas Expertos.

El objetivo de una entrevista no solo incorpora el contenido de una discusión, sino también la precisión con la cual los participantes deben comunicarse y las relaciones entre el entrevistador y el entrevistado. En este capítulo se describe cómo el objetivo de la creación de un Sistema Experto influencia las entrevistas de Adquisición del Conocimiento y cómo la Adquisición del Conocimiento difiere de otras actividades de la entrevista. [14]

El contenido de una entrevista de Adquisición del Conocimiento involucra el proceso por el cual el experto realiza la tarea, las razones que están detrás de cada acción y los hechos y decisiones del experto necesarios para realizar la tarea seleccionada.

En la mayoría de los diálogos existen algunas discrepancias entre el significado que desea expresar el entrevistado y el que entiende el entrevistador. La comunicación suele ser más efectiva cuando ambos comparten el mismo lenguaje, cultura, educación, puntos de vista y familiaridad con el problema o tarea. En este sentido es donde el Ing. del Conocimiento y el experto están en gran desventaja ya que la naturaleza de sus trabajos es por lo general muy diferente. En consecuencia, es posible que inicialmente el Ing. del Conocimiento no sepa cuales son las preguntas adecuadas y por ello el experto tenga dificultades para entenderlas y responderlas.

Cuando el Ing. del Conocimiento y el experto comienzan a trabajar en un proyecto, su comunicación puede no ser efectiva. Los malos entendidos pueden ser grandes debido a la diferencia de experiencias y educación, por lo cual es importante establecer un vocabulario común y revisar con cierta frecuencia que el Ing. del Conocimiento está entendiendo correctamente lo que el experto desea significar.



Adicionalmente a esto, es probable que durante la entrevista se pueda apreciar que el experto no este consiente de su conocimiento o de como lo utiliza, probablemente sea la primera vez que tenga que explicarlo a detalle por lo que se debe tener en cuenta que el ser forzado a decir cosas o a pensar acerca de estas, puede afectar la manera en la que el experto ve el proceso.

Estos problemas pueden ser aparentemente triviales, pero pueden ocasionar grandes conflictos en los resultados finales por lo cual se consideraran a lo largo del capítulo proponiendo algunas soluciones.

A continuación se presenta un esquema de como una entrevista debe ser dirigida por el Ing. del Conocimiento (Fig 3.1):

## ENTREVISTA

NIVEL: PARTICIPANTES		
Interaccion		
Experto	Ingeniero del Conocimiento	
	Uno	Múltiples

Fig 3.1 Esquema de la Entrevista

NIVEL: PLANEACION Y PREPARACION				
Contenido	Tiempo y Lugar	Material	Experto	Ingeniero del Conocimiento

NIVEL: TRANSFERENCIA DE INFORMACION					
Comunicacion Verbal			Captura de Informacion		
Introduccion	Cuerpo	Final	Notas	Pizarron	Audio y Video Grabaciones

NIVEL: DIALOGO				
Introduccion del Tema	Preguntas	Respuestas		Cerrar el Tema
		Escuchar	Responder	

### I.I. Participantes.

Cada participante en una entrevista juega un rol en particular y la interacción apropiada entre los participantes es crucial para el éxito de la entrevista.

Una entrevista simple tiene únicamente dos participantes: un Ing. del Conocimiento y un experto. El Ing. del Conocimiento juega el rol de entrevistador y el experto el rol de entrevistado. Existen muchos proyectos en los que son incluidos varios Ing. del Conocimiento. Cuando estos equipos de ingenieros entrevistan al experto, estos deben trabajar juntos de manera que no dificulten la labor del experto.

#### **Interacción entre el Ing. del Conocimiento y el experto.**

La manera en la cual el Ing. del Conocimiento interactúa con el experto durante una entrevista es de vital importancia para el proyecto, pues a través de esta interacción es que se establece una buena relación de trabajo. Las primeras interacciones son particularmente importantes. Si el o los Ing. del Conocimiento comienzan en malos términos con el experto, puede que posteriormente sea imposible reparar el daño. Es de gran importancia que durante la entrevista, el experto se sienta a gusto.

Cada experto es un individuo diferente, por lo que la mejor manera de interactuar con uno puede no serlo con otro. Por ello, es necesario ser sensitivo con el experto y adaptarse a su estilo, por lo que es importante incluirlo en todas las discusiones y tratarlo en una manera que demuestre respeto y paciencia con el objetivo de que se sienta parte del equipo y aumente su entusiasmo.

También es importante considerar la comunicación no verbal con el experto. El dialogo es solo una parte de la comunicación. La comunicación efectiva requiere que se note, entienda y responda al experto con mensajes no verbales. Las expresiones faciales, las pausas, la velocidad a la que se habla y los movimientos de las manos y cuerpo, también transmiten información. Es importante poner atención a estos mensajes, decidir que significan y responder apropiadamente.

Por lo antes mencionado, es importante familiarizarse con el estilo de comunicación con el experto. Durante las primeras entrevistas, se debe intentar aprender lo que es normal para el experto y buscar patrones de comportamiento que indiquen que todo va bien. Estos patrones incluyen el flujo normal de

conversación del experto, expresiones faciales comunes y gestos frecuentes. Las variaciones de tales patrones pueden indicar que la entrevista no va por buen camino. A medida que se tenga familiaridad con el estilo de comunicación del experto, se podrán identificar desviaciones con los patrones normales y aprender a interpretar tales desviaciones.

Es importante discernir las **señales** que indican que el experto está cansado, frustrado, entusiasta, inseguro de la información, preocupado o complacido y tomar las acciones apropiadas. Por ejemplo, si el experto se ve cansado o distraído se debe tomar un descanso y considerar en finalizar la sesión temprano. Si se ve frustrado con la línea actual de cuestionamiento, o se le dificulta el tópico que se está tratando, se debe tomar un descanso o cambiar el tópico y posteriormente, tal vez en otra sesión, regresar a la discusión actual. Si el experto se ve inseguro de una respuesta o se ve menos confidente de lo usual acerca de alguna información, se debe preguntar al experto si desea postergar la discusión detallada para cuando disponga de más tiempo. También es importante poner atención a las reacciones del experto cuando otras personas están presentes en la entrevista, si reacciona negativamente, se debe tratar de limitar el número de personas. Es imprescindible detectar los sentimientos positivos para incentivarlos y los negativos para tratar de corregir las causas.

Si en algún momento durante la entrevista el Ing. del Conocimiento se siente cansado, aburrido, frustrado o impaciente y lo demuestra al experto, puede significar que sienta que la entrevista es improductiva y que está perdiendo tiempo valioso. Si el Ing. del Conocimiento tiene cierto tipo de problemas personales, familiares o de trabajo, es posible que envíe ciertas señales al experto que le hagan sentir que tiene problemas con él, en este caso lo recomendable es explicar la situación de manera que el experto no conserve una impresión equivocada. Finalmente, el Ing. del Conocimiento debe asegurarse que sus señales no-verbales sean congruentes con las verbales. Si este envía al experto mensajes mezclados, puede provocar que el experto ponga atención a sus acciones y no a sus palabras.

Finalmente, se deben considerar de igual importancia las **diferencias culturales**. El Ing. del Conocimiento debe estar consciente de ellas y corregir o ajustar su comportamiento. Lo que puede ser normal, rutinario y cómodo para uno puede ser muy desagradable para el otro. Es responsabilidad del Ing. del Conocimiento notarlo y acomodarse a las preferencias del experto adaptándose a sus normas de formalidad, vestuario y espacio personal.

### **Interacción con múltiples Ing. del Conocimiento.**

Si las entrevistas son de alguna manera intimidantes para el experto, las entrevistas con todo un equipo pueden ser peores aún. Los problemas podrían ser mayores a medida que crece el equipo, por lo que se debe limitar el número de personas que pregunten en una entrevista. Como regla general, deberían ser uno o dos los Ing. del Conocimiento que realicen las preguntas y permitir que ocasionalmente intervenga el resto. Si se detecta que el experto está extrañamente callado o inquieto, se debe limitar el equipo de entrevistas a dos personas. Si aún así persiste tal actitud, debe limitarse a encomendar la tarea a un solo Ing. del Conocimiento.

Se debe tener el cuidado de no arrollar al experto, es decir no ir demasiado rápido ni agobiarlo. Es importante darle tiempo para descansar. Se debe recordar que mientras los Ing. del Conocimiento hacen preguntas alternativamente, el experto debe responder a todas. Lo que puede ser confortable para el equipo de ingenieros, puede ser terriblemente cansado para el experto.

Finalmente, antes de formular una pregunta se debe permitir al experto responder a la previa. Cuando varios Ing. del Conocimiento repiten la misma pregunta con diferentes palabras sin esperar a una respuesta, se da la impresión de que se está presionando al experto.

### **Interacción entre los Ing. del Conocimiento.**

Durante las entrevistas, la interacción entre los Ing. del Conocimiento debe ser cordial y cooperativa. Como el nombre de equipo de entrevistas lo indica, los Ing. del Conocimiento que participan en la entrevista deben trabajar juntos en equipo. Si más de una persona realiza las preguntas, el realizar trabajo de equipo es particularmente importante. Cada pregunta debe seguir de manera natural a la precedente y su respuesta como si una sola persona estuviera realizando todas las preguntas.

Si surgen diferencias de opinión, estas debe ser discutidas de manera amable. El entrevistador líder debe decidir qué alternativa es la más recomendable y el resto de Ing. del Conocimiento debe respetar esta decisión. Si fuese necesario, los Ing. del Conocimiento pueden discutir sus desacuerdos después de la sesión cuando el experto no esté presente. Este tipo de discusiones pueden disminuir la confianza que el experto tenga en el proyecto.

### I.II. Planeación de una entrevista.

Esta sección describe el proceso mediante el cual el Ing. del Conocimiento puede preparar una entrevista.

A continuación se presenta una lista de los factores a considerar como una guía para la preparación del plan a seguir:

- Planificación del contenido de la entrevista.
- Planificar el tiempo y el lugar.
- Identificar el material necesario.
- Preparar al experto.
- Preparación del Ing. del Conocimiento.

#### **Planificación del contenido.**

El primer paso en la planificación de la entrevista, es la identificación de los temas a tratar. Se debe establecer tan específicamente como sea posible qué es lo que se desea obtener durante la entrevista. Revisar las metas a cubrir en el estado actual de la Adquisición del Conocimiento, e identificar las relevantes para la entrevista actual. Se debe estar seguro de que los temas de la discusión direccionan todas las metas relevantes. Durante las primeras entrevistas, se debe prestar especial atención a las metas propuestas para la conformación del equipo de trabajo. En las entrevistas posteriores, debe concentrarse en las metas de recolección de la información.

Posteriormente, el Ing. del Conocimiento debe identificar las suposiciones y prejuicios acerca de los temas que se planeó discutir. Durante la entrevista, se debe evitar introducir opiniones propias. Si el tema de discusión implica suposiciones, se deben incluir nuevos temas con el fin de saber si las suposiciones son correctas.

Una vez que se decidió qué información se espera obtener, se debe listar las actividades que serán útiles en la adquisición de la información para posteriormente crear una agenda a partir de la lista de actividades seleccionadas, y estimar la cantidad de tiempo que llevará cada actividad. El propósito de la agenda es revisar que el contenido sea razonable dado el tiempo que se dispone. La agenda debe ser una descripción flexible de los objetivos, no un plan rígido. Al principio del proyecto, los estimados de tiempo pueden no

ser seguros. Antes de conocer al experto no se poseerá un buen sentido de cuanto tiempo se le puede dedicar a las diferentes actividades. Más adelante en el proyecto se podrán hacer estimados más seguros, pero aún así se debe estar preparado para desviarse del plan.

#### **Planificar el tiempo y el lugar.**

Es conveniente para todos los participantes establecer el tiempo y la localización de las entrevistas. Si es posible, al finalizar cada una de ellas se debe decidir sobre el tiempo que se empleará en la siguiente.

A medida que el proyecto progrese, será posible planificar tiempos estándares (por ejemplo: los Miércoles de 1:00 a 5:00 p.m.). De manera similar, es posible llegar a un arreglo para que todas las entrevistas tengan lugar en el mismo sitio. Un arreglo ideal sería que todas las juntas se lleven a cabo en una sala que esté reservada para el proyecto, de manera que los materiales de referencia comunes, los cuadernos del proyecto, etc. se puedan dejar en la sala de proyectos.

#### **Identificar el material necesario.**

Una vez que ha sido revisada y verificada la agenda, el Ing. del Conocimiento debe especificar que materiales necesitará en la junta. Los materiales de grabación son de gran utilidad, así como los pizarrones. Es importante que cuando el Ing. del Conocimiento considere que tipo de materiales necesitará, tome en cuenta que es lo que desea mostrar al experto, que desea que el experto le muestre y que material le sería útil para su discusión.

Si la lista incluye materiales escritos para ser discutidos, es imprescindible tener una copia para cada persona que intervenga en la junta, y de ser posible que este material se distribuya antes de la entrevista.

#### **Preparar al experto.**

Para que una entrevista sea productiva, es importante que todos los participantes estén preparados. El Ing. del Conocimiento debe asegurarse que el experto llegue preparado a las entrevistas. Debe asegurarse de que el experto sepa la hora y localización de la junta. Si fuese necesario recordarle que materiales debe llevar y ver si este tendrá el

tiempo suficiente para prepararlo antes de la entrevista. Si el experto no estuviese preparado, se debe considerar el reprogramar la entrevista para otro día o trabajar en algún tema alternativo. Al principio del proyecto o cuando las juntas son muy esporádicas, el Ing. del Conocimiento deberá proporcionar más información y ofrecer más ayuda.

#### **Preparación del Ing. del Conocimiento.**

Es igualmente importante que antes de la entrevista, el Ing. del Conocimiento revise su agenda y notas de las sesiones anteriores, de manera que los asuntos importantes relacionados con los temas a discutir en la junta estén al día. Si se tiene material escrito de los asuntos a tratar en la entrevista, estos deben ser revisados con la finalidad de preparar comentarios o preguntas si fuesen necesarios.

#### **I.III. Transferencia de información.**

La transferencia exitosa de información durante una entrevista de Adquisición del Conocimiento requiere que exista una buena Comunicación de información del experto al Ing. del Conocimiento y que este último almacene la información para su posterior revisión.

#### **Comunicación verbal.**

Como se mencionó con anterioridad, gran parte de la comunicación verbal en una entrevista de Adquisición del Conocimiento ocurre mediante un proceso que consiste de una serie de preguntas y respuestas. La comunicación será más productiva si las preguntas presentan una secuencia lógica de temas.

A pesar de que generalmente los Ing. del conocimiento son los que establecen la dirección de la discusión al hacer las preguntas, el experto debe ser capaz de entender y seguir tal dirección. El experto debe conocer las metas que el Ing. del Conocimiento tiene para una entrevista, entender el tema al que está direccionada cada pregunta, reconocer cuando la discusión deja un tema y procede con el siguiente, entender como están relacionados los temas sucesivos y porque están incluidos en la misma entrevista, y al final de la entrevista, saber que esperar en el futuro.

El Ing. del Conocimiento debe estructurar las entrevistas para facilitar el proceso al experto y mantenerlo orientado hacia el tema actual de la discusión. Cada entrevista debe consistir de tres segmentos:

- Una introducción.
- Un cuerpo.
- Un final.

#### Introducción.

Se debe comenzar cada entrevista con una introducción informal en la que se hará conocer al experto lo que se tiene planeado. En varias de las primeras entrevistas, el Ing. del Conocimiento necesita explicar al experto cuales son los objetivos que se están tratando de alcanzar y que es lo que espera del él. En las subsecuentes entrevistas, la introducción deberá incluir un sumario breve de lo que se discutió en la junta previa y que es lo que el Ing. ha logrado desde entonces.

#### Cuerpo de la entrevista.

Durante el desarrollo de una entrevista, la secuencia de temas debe fluir de manera suave. Todos los participantes deben saber cuando una discusión deja un tema y procede con el siguiente. Si se sigue la agenda de la entrevista, la secuencia de temas será mas clara para el experto. A pesar de ello, muchas entrevistas se desvían de la agenda planeada. Esto puede ser causado porque la técnica de cuestionamiento no es buena, porque la discusión de un tema requiere mas tiempo que el esperado, porque el experto puede no estar preparado, o porque la discusión de un tema puede identificar otros que deben ser cubiertos en la entrevista actual. Este tipo de desviación es común al principio del proyecto cuando el Ing. del Conocimiento aún no conoce lo suficiente sobre el experto para planificar exactamente cuales son los temas que se deberían discutir.

#### El final de la entrevista.

El final de una entrevista es tan importante como la introducción. La finalización puede asegurar que todos los miembros del equipo consideren la discusión completa. Para cerrar una entrevista se debe:



1. Dar a todos los participantes la oportunidad de hacer comentarios finales acerca del tema tratado o de proponer temas para futuras entrevistas.

2. Sumarizar lo que el grupo ha logrado en la entrevista. Identificar rápidamente los temas de la agenda que fueron discutidos completamente, los que fueron parcialmente discutidos, aquellos que no fueron discutidos y los temas no planeados que fueron discutidos.

3. Llegar a un acuerdo sobre una lista tentativa de temas de discusión para la siguiente entrevista.

4. Identificar la preparación que cada participante necesitará antes de la siguiente junta.

5. Establecer los horarios para la siguiente junta (si fuese necesario).

Antes de abandonar la junta se debe agradecer al experto por su tiempo en la preparación y participación en la entrevista.

#### **Como capturar la información de una entrevista.**

En una entrevista exitosa, el Ing. del Conocimiento obtiene del experto información importante que será utilizada para diseñar e implementar el Sistema Experto. El ingeniero no debe confiar en su memoria para almacenar toda esta información, ya que si olvida algún detalle relativamente menor, es posible que el Sistema Experto no funcione correctamente. Si el Ing. del Conocimiento necesita repetir la discusión con el experto, pierde tiempo valioso del proyecto e inspira la desconfianza del experto en sus habilidades. El Ing. del Conocimiento puede almacenar la información tomando notas, utilizando pizarrones o grabando la entrevista en cinta o en video.

#### **Notas.**

La manera más común de registrar una comunicación verbal es a través de notas. El Ing. del Conocimiento debe tener su cuaderno de proyectos para plasmar ideas importantes, definición de términos nuevos, y preguntas para futuras discusiones. Las notas capturan más de lo que el experto dice, estas pueden registrar lo que Ing. del Conocimiento piensa acerca de :

- Que información es especialmente importante?.
- Que temas deben ser discutidos en la entrevista actual?.
- Que temas deben ser explorados con el experto en entrevistas futuras?.
- Que temas deben ser discutidos entre los Ing. del Conocimiento (en ausencia del experto)?.

La toma de notas es la manera mas condensada y versátil de capturar la información de un experto. Desafortunadamente no es el método más confiable de obtener los detalles de una discusión.

### Pizarrones.

Los pizarrones son útiles para escribir ideas, puntos importantes o diagramas. Además, muchos expertos se sienten mas seguros cuando van escribiendo sus ideas en un pizarrón a medida que las van exponiendo. El problema que se presenta con estos medios es que si se desea guardar la información escrita sobre ellos, esta debe ser copiada a mano.

### Audio y Video grabaciones.

Para muchos Ing. del Conocimiento es imprescindible grabar las entrevistas para poder complementar sus notas. Las grabaciones son particularmente útiles cuando un miembro del equipo del proyecto falta a alguna sesión o si se cuenta únicamente con un Ing. del Conocimiento que tiene que conducir la entrevista y al mismo tiempo tomar las notas. Por otra parte, el experto podría estar mencionando información que el Ing. del Conocimiento no oye o no presta atención debido a que no encaja en el modelo actual del problema. Cuando el modelo cambia, se pueden producir cambios radicales y se necesitará revisar fragmentos de conversaciones pasadas con el experto.

### I. IV. Dialogo.

En esta sección se presentan algunas técnicas para la conducción de un dialogo en una entrevista. El Ing. del Conocimiento debe seguir una secuencia de 5 actividades durante la entrevista. Estas actividades son:

1. Introducir un tema.
2. Realizar una pregunta.
3. Escuchar la respuesta.
4. Responder a la respuesta.
5. Cerrar el tema.

que a continuación se detallan.

#### **Introducción de un tema.**

La estructura de un dialogo dentro de una entrevista, ayuda a orientar al experto hacia las metas de la Adquisición del Conocimiento. El Ing. del Conocimiento debe introducir cada nuevo tema de discusión con el propósito de establecer el enfoque actual ante el experto y los demás Ing. del Conocimiento si fuese necesario, y para asentar la relevancia de las preguntas que se realizarán.

La introducción debe establecer la importancia del tema elegido. Es posible que el Ing. del Conocimiento tenga que explicar que es lo que se está tratando de encontrar, y porque esta información es importante para la construcción del Sistema Experto.

#### **Realización de preguntas.**

Después de la introducción de un tema, el Ing. del Conocimiento debe hacer las preguntas relacionadas con el tema. La realización de una pregunta puede estar conformada por cuatro pasos:

1. El primer paso en la formulación de una pregunta es identificar un objetivo: el conocimiento que se desea aprender del experto.

2. Después de identificar lo que se desea preguntar, se debe decidir como realizar la pregunta. Para poner el requerimiento de información en palabras es necesario:

- Seleccionar el tipo apropiado de pregunta. Se deben formular preguntas abiertas cuando se requiere información general y preguntas más cerradas si se busca detalles específicos. También se debe poner atención a las reacciones del experto ante las preguntas ya que algunos tiene serias dificultades para responder preguntas muy abiertas y otros se sienten amenazados ante preguntas muy cerradas.

- Expresar la pregunta de manera que sea tan clara como sea posible, para lo cual se deben seleccionar palabras que el experto entienda, realizar preguntas cortas y simples evitando cláusulas complicadas y si es necesario explicar el contexto de la pregunta, y evitar ambigüedades de manera que cada pregunta tenga una sola interpretación. Para eliminar tales ambigüedades se debe evitar utilizar los pronombres "esto, aquello y eso" pues su significado puede no ser claro.

3. Una vez que se ha puesto la pregunta en palabras, se deben pronunciar las palabras para que el experto las oiga. Es de gran importancia cuidar el tono de voz. Se debe hablar claro, enunciar con cuidado y asegurarse de no hablar muy rápido con el propósito de que el experto entienda claramente.

4. Una pregunta que no obtiene una respuesta, es fallida. El Ing. del Conocimiento debe dar el tiempo suficiente para que el experto responda una pregunta, si aún así no la obtiene, debe detectar la falla tan pronto como le sea posible y corregirla. Las preguntas pueden fallar por muchas razones:

- Si la pregunta es muy general, es posible que tenga muchas respuestas y que el experto no pueda decidir cual es la respuesta correcta al no poder determinar el significado exacto de la pregunta. Si se detecta que el experto no está preparado para responder a preguntas muy generales, el Ing. del Conocimiento debe formular preguntas más específicas pero evitando direccionarlas basándose en su propio conocimiento, suposiciones o prejuicios ya que se puede caer en otra pregunta fallida.

- Es posible que el experto no entienda la pregunta y lo exprese de esta manera, pero por lo general tratará de figurarse su significado y responder a este último. Para solucionar este problema, el Ing. del Conocimiento debe tratar de reformular mas claramente la pregunta.

- También es posible que el experto no entienda la relevancia de la pregunta y se muestre renuente a cooperar. Este caso puede presentarse cuando no sea clara la conexión entre la introducción a un tema y la pregunta.

- Finalmente es posible que el experto no sepa la respuesta. Si este es el caso, se debe hacer saber al experto que la discusión se pospondrá a una entrevista posterior de manera que este tenga el tiempo suficiente para investigar la respuesta.

**Escuchar la respuesta del experto.**

Después de haber realizado una pregunta, es necesario poner atención a la respuesta del experto. No se puede obtener información sino se escucha lo que el experto tiene que decir. Se debe poner mucha atención a los problemas que interfieran con el enfoque del Ing. del Conocimiento en la respuesta del experto. A continuación se mencionan algunas de las causas que generan tales problemas.

- Frecuentemente los Ing. del Conocimiento inexpertos están ansiosos por demostrar su habilidad de mantener en movimiento la discusión. Tan pronto como formulan una pregunta, ya están haciendo la otra sin darle oportunidad al experto a que responda con libertad y a sí mismos a asimilar la respuesta.
- Algunas veces los Ing. del Conocimiento se distraen con movimientos, escenarios o sonidos externos a la entrevista.
- Es posible que en una entrevista demasiado larga, el Ing. del Conocimiento esté demasiado cansado para concentrarse en la entrevista y no preste atención a las explicaciones del experto.
- Algunos expertos proveen mucha información demasiado rápido, de manera que el Ing. del Conocimiento no alcanza a comprender todo. A menudo los Ing. del Conocimiento inexpertos en lugar de sugerir al experto que vaya más despacio, les provoca pánico y dejan de escuchar.

Cuando un experto responde una pregunta, se debe escuchar y tratar de comprender. Es muy importante entender lo que el experto quiere decir, no solo oír y registrar lo que dijo el experto. El que el Ing. del Conocimiento tome notas, le pueden ayudar a juzgar la completez de su entendimiento. Si no está seguro de lo que escribió o las notas no dicen todo lo que se desea saber, es necesario formular otras preguntas para obtener información adicional. En este sentido el Ing. del Conocimiento debe concentrarse en el contenido de las notas no en la forma, con el propósito de que no interfieran con su atención a las respuestas.

Antes de comenzar a responder al experto, el Ing. del Conocimiento debe estar seguro de haber identificado todos los aspectos de la contestación que requiere una respuesta. Analizar la completez y claridad de la respuesta y la

completez de su propio entendimiento. Revisar si la respuesta encaja con su percepción actual o se contradice con la aprendido en las entrevistas anteriores.

#### **Responder a la respuesta.**

Siempre se debe responder a la contestación del experto antes de formular la siguiente pregunta. Este proceso no solo mejora el entendimiento del Ing. del Conocimiento, sino que demuestra al experto que se le está poniendo atención.

Existen varias técnicas que pueden ser utilizadas para asegurar que el Ing. del Conocimiento tuvo un entendimiento claro y completo de la respuesta del experto. Estas técnicas también incentivan al experto a continuar o expandir un tema.

Estas técnicas están divididas en cinco tipos de respuesta:

1. Formulando las subsecuentes preguntas. Si el Ing. del Conocimiento piensa que tiene una respuesta incompleta, superficial, inexacta o vaga, formula otras preguntas.
2. Confirmar su entendimiento. Se presenta cuando el Ing. del Conocimiento cree haber entendido la explicación y repite al experto en sus propias palabras lo que le pareció importante. Esto da la oportunidad al experto de corregir una concepción equivocada.
3. Resolver las contradicciones. Si la respuesta parece inconsistente con el entendimiento actual del Ing. del Conocimiento, es necesario resolver tales contradicciones haciendo saber al experto que no se tiene la seguridad de cual opción es la correcta, o cuales son las condiciones bajo las que cada alternativa es la apropiada.
4. Aprender la terminología del experto. Cuando el Ing. del Conocimiento escucha un término que no le es familiar, debe solicitar al experto una definición del significado de tal palabra.
5. Modificar la agenda. Cuando la respuesta sugiere temas adicionales y preguntas, la agenda debe ser actualizada. Si la respuesta del experto produce nuevas dudas, no se debe interrumpir al experto en su tren de pensamiento, en su lugar

se debe anotar las preguntas y dudas para aclaralas en el momento adecuado.

**Cerrar el tema.**

Después de que el Ing. del Conocimiento ha formulado todas las preguntas necesarias en un tema, debe cerrarlo. Este hecho señalará al experto que se terminó con el tema presente y que se está listo para continuar con el siguiente. Es importante recordar preguntar las definiciones de los términos nuevos, sumarizar los puntos importantes y conservar juntas las ideas discutidas. Antes de abandonar un tema, se debe confirmar con el experto que toda la información importante y disponible ha sido obtenida.

## II. ANÁLISIS DE PROTOCOLO.

El análisis de Protocolo, está basado en la transcripción de una entrevista: su objetivo es tratar de estructurar el proceso y produce resultados más significativos. Este método ha sido utilizado con éxito en la criptometría, diagnósticos médicos, y diseño de componentes mecánicos.

Como ya se mencionó con anterioridad, para los expertos es más fácil hablar sobre ejemplos de problemas específicos que hablar en términos abstractos. Para ellos es más fácil responder a preguntas como: ¿Cómo sabe que este diseño no funcionará? en lugar de ¿Qué hace que un diseño sea pobre o malo?. El experto toma casos documentados, es decir reportes de laboratorio o historiales de casos y habla acerca de estos, es decir como si pensara en voz alta. Si durante este proceso el experto se concentra en una serie de casos específicos, significa que no le agrada apartarse de un tipo particular de casos, ni estar brincando de un caso a otro, lo cual en general hace que sus comentarios sean más coherentes y estructurados. De los comentarios de un ejemplo específico es posible detectar patrones generales, es decir es posible que el experto contemple siempre una característica en particular antes que otras. Si se presenta este caso, es importante considerar que es más fácil estructurar el conocimiento en grupos y conceptos.

El método de Análisis de Protocolo propone que además de todas las sugerencias para una entrevista desarrolladas en la sección anterior, es necesario y recomendable hacer una transcripción de todos los comentarios y analizarlos. La principal ventaja de este método es que las transcripciones describen los comentarios del experto durante su etapa de trabajo, es decir hablando sobre su toma de decisiones a medida que las toma.

El análisis de las transcripciones requiere un particionamiento sistemático de la información para producir un modelo estructurado del conocimiento del experto.

En el primer estado del desarrollo de este protocolo se identifican las oraciones con mayor contenido de información de la transcripción. La recolección de estas oraciones es simplificada y re-escrita para construir un modelo simple de conocimiento. Las oraciones ya refinadas son reunidas en una tabla de oraciones "Si...Entonces" las cuales forman la base



del análisis. Esto identifica las áreas de conocimiento y las relaciones entre las diferentes áreas. Las reglas son utilizadas para producir una serie de modelos. En cada estado, el modelo es probado, modificado y entonces es almacenado como fijo. Estas modificaciones consisten de adiciones, eliminaciones o mejoras de arreglos de reglas basadas en los resultados de las pruebas y los consejos del experto. Los modelos sucesivos deben ser mejores, aunque es posible que exista una tendencia a sobre-especificar.

Los resultados posteriores a este proyecto muestran que es difícil tener un grado de seguridad perfecto en un diagnóstico pues los problemas más graves tienen una estructura completamente diferente a la de los problemas comunes. Otro problema que está presente en la mayoría de los Sistemas Expertos, es su poca habilidad para reconocer los límites de su propia capacidad, es decir, es raro obtener una respuesta "no se" a una consulta que se sabe que está fuera de su capacidad de resolución. [3].

### III. ADQUISICIÓN DEL CONOCIMIENTO DE MÚLTIPLES EXPERTOS.

A menudo los proyectos de Sistemas Expertos están basados en la colaboración de un solo experto del dominio. La dificultad que se presenta en estos casos es que no se puede estar seguro de que tan adecuados son los datos y la tarea elegida por un solo experto. A continuación se presentan algunas ventajas al utilizar un conjunto diverso de expertos.

Un problema que se presenta con mucha frecuencia en la construcción de los Sistemas Expertos radica en la determinación del dominio. Esto se debe principalmente a que no es sencillo identificar a los expertos y el tipo de problemas que resuelven, por lo cual se debe adquirir y verificar la naturaleza y el alcance de la expertez antes de construir un Sistema Experto.

En dominios complejos y variados como la medicina o la Ingeniería donde un experto posee conocimientos profundos acerca de un subconjunto pequeño de tareas en el dominio, se considera más apropiado que un grupo de expertos sugieran cual es la tarea más adecuada para realizar el modelado del sistema.

Es posible que un solo experto (a menudo teórico) sugiera una tarea que sea de gran interés para él y que sustente su elección en el hecho de tener en general problemas con dicha tarea, muchos casos en los que se presente, y gran cantidad de información organizada de los estudios realizados al respecto. Sin embargo es posible que al ser presentado a otros expertos o gente experimentada en el área, no posea la aportación esperada, no proporcione nada que no se pueda realizar con una rutina básica y que muchos de estos prefieran un sistema que los ayude a realizar tareas con las cuales tienen dificultad.

Por lo anteriormente expuesto se debe estar seguro que los expertos sean "expertos prácticos" en la tarea seleccionada. Además si se esta planeando construir un Sistema Experto para auxiliar a expertos humanos, se debe estar seguro de qué es en lo que necesitan ayuda. Es decir identificar los usuarios apropiados antes de construir el sistema.

Es muy común que en una misma área de expertez existan varios expertos que tengan un enfoque y actividad diferente dentro de la misma área, por lo cual si solo se tomara a uno de ellos, su enfoque (y por lo tanto el sistema) sería de poca utilidad para los otros. Por el contrario si se toma una variedad de ellos al principio del proyecto, es posible tener un mejor entendimiento de los diferentes tipos de expertez que prevalecen en el dominio y cual es el tipo de conocimiento en el que trabaja cada experto.

A continuación se establece un formato para entrevistar a varios expertos de diferentes tipos.

#### Entrevista a múltiples expertos.

La entrevista debe ser conducida por el Ing. del Conocimiento pero usando como colaboradores a expertos residentes.

-Los expertos residentes pueden presentar sus problemas de diseño y después realizar básicamente labores de consultores sobre el problema aunque no participarán en el proceso de diseño.

-El mismo problema debe exponerse a cada uno de los expertos sin la presencia de otro.

-A los expertos se les pedirá que participen en el proceso de diseño aunque no necesariamente haciendo diseño detallado, si no todo lo que puedan hacer en aproximadamente dos horas.

-Es importante enfatizar que el Ing. del Conocimiento no pedirá a los expertos que le expliquen como resuelven problemas de diseño, sino que realicen un diseño de prueba en base a ciertas especificaciones.

-No se debe esperar que los expertos puedan dar una cantidad de su conocimiento. El Ing. del Conocimiento debe ejercitar la expertez de los expertos en problemas reales para extraer y modelar su conocimiento.

Estas entrevistas deben seguir los objetivos que a continuación se describen.

### ¿Que se aprenderá de las entrevistas?

Se desea que los expertos residentes proporcionen los siguientes tipos de información antes de hablar con los otros expertos.

**- Solución del Problema:**

Se les pide que "resuelvan" problemas de diseño del trabajo que realizan otros expertos. Esto permite al Ing. del Conocimiento formar un modelo inicial de como estos diseños fueron realizados en términos de algunos de los pasos iniciales.

**- Identificación de subproblemas:**

Estos también deberán identificar uno de los subproblemas del diseño como un buen candidato para la construcción del Sistema.

**- Verificación por comprobación:**

Finalmente se examina cómo otros cinco expertos trabajan el mismo problema. Los resultados muy probablemente sean sorprendivos y alentadores.

### Similitud en las aproximaciones.

Todos los expertos siguen una estrategia similar en la realización del diseño en términos de como descomponen el problema en subproblemas, como trabajan en los subproblemas y como relacionan los diseños parciales unos con otros.

Es importante poder apreciar que en el área del problema seleccionado se tienen algunas regularidades en las que cada experto ha sido entrenado para explotar. Al finalizar el Ing. del Conocimiento podrá desarrollar un protocolo inicial para el diseño del subsistema elegido.

### Diferencias en la especialización.

Aunque diferentes expertos siguen un protocolo similar en el diseño, es claro que existen subáreas de expertez en las cuales se pueden especializar, inclusive algunas de estas en las que los expertos residentes no saben mucho, o en algunas otras áreas de especialización que surgen de las entrevistas como son: análisis de costos, recopilación de especificaciones muy herméticas y uso de tecnologías alternativas.

Estas observaciones son muy importantes debido a las siguientes razones:

- Dado que existe un nivel de expertez compartido, tiene sentido tratar de capturarlo como un sistema experto.
  
- Existen niveles de conocimiento especializado que pueden ser vistas como conocimiento común, de donde se puede tratar de capturar una Base de Conocimientos común.
  
- Dado que existen diferentes especialistas, se puede hablar con ellos con más profundidad a medida que vaya siendo necesario.

#### **Naturaleza de la expertez.**

Aunque diferentes expertos tiene un entendimiento similar del dominio de la tarea, existen diferencias en la naturaleza de su expertez. Se presupone que esto está relacionado con la cantidad de veces que la expertez fue asimilada y en que fue asentada.

Algunos expertos pueden realizar un diseño sin la necesidad de analizar a fondo el problema aunque sea nuevo para ellos. Para otros es necesario realizar un análisis más profundo y a menudo necesitan hacer revisiones, y regresar desde el principio para reafirmar. Estos últimos por lo general son los que no han trabajado con este tipo de problemas antes.

La diferencia respecto a las bases sobre la que asientan su expertez se presenta cuando el Ing. del Conocimiento realiza preguntas sobre el por que realizan un paso dado, por que siguen ciertas reglas y por que algunos pasos obligatorios son más importantes que otros. Las repuestas que por lo general se obtiene son que ellos siempre han realizado algunos procesos de cierta manera y con buenos resultados, mientras que otros necesitan revisar extensivamente las bases teóricas antes de tomar alguna decisión de diseño.

### **Separación de tareas.**

Algunos de los subproblemas identificados por los expertos residentes pueden estar muy involucrados con otros. Es posible que ninguno de los expertos haya trabajado en el subproblema de manera aislada dado que este es demasiado dependiente del resto del diseño. El grado de interdependencia muestra claramente que se debe considerar el problema completo o no considerarlo en absoluto.

### **Ejercitar la expertez.**

Por lo general existe mucha gente trabajando en el mismo problema ayudando a llenar huecos de la especificación que los expertos residentes pudieron haber preparado para el panel de expertos. Es de esperar que los otros pregunten acerca de la información que no está presente al pensar que olvidaron incluirla o que al estar tan habituados a trabajar con tales subproblemas, no les es posible enumerar fácilmente todos los aspectos. Con este hecho se ejercita la expertez lo cual es necesario para obtener información de ellos. Esta experiencia es muy importante cuando finalmente se obtiene información detallada de los expertos residentes ante las preguntas e inquietud del resto.

### **Discusión.**

Una de las áreas de actividades de los Sistemas Expertos menos tratadas es la parte inicial, cuando se decide respecto a que es lo más adecuado del problema y a la identificación de los expertos. Esta fase es tan importante como las de decisión de las técnicas de representación del conocimiento, estrategias de resolución de problemas o tipo de programación.

Una técnica útil es el proceso de entrevista sistemática a un grupo diverso de expertos en un área tentativa de problema y la realización de algún protocolo de análisis de sus actividades en la resolución de problemas típicos. Muchas de las preguntas que afectan directamente los resultados de un proyecto pueden ser respondidas al entrevistar a muchos y variados expertos.

Algunas de estas preguntas se discuten a continuación:

- + Identificar el problema y los expertos.  
 ¿Quiénes son los expertos y en qué trabajan generalmente?  
 ¿Quiénes son el o los expertos de los cuales se va a adquirir el conocimiento detallado?  
 ¿Estos están reconocidos como valiosas fuentes de conocimiento en su medio?

El hablar con un grupo diverso de expertos ayudara a determinar la reputación, en su comunidad, de los expertos que están colaborando.

- + Identificar a los usuarios.  
 ¿Están de acuerdo los expertos sobre la importancia del problema ya sea del punto de vista de que hará su conocimiento más accesible o que les ayudará a realizar mejor su trabajo?  
 ¿Los expertos están dispuestos a invertir recursos en la construcción del sistema?

Es importante no solo identificar a los usuarios de Sistema Experto, sino también juzgar su compromiso en términos de cuanto están dispuestos a invertir. Si no están dispuestos a invertir nada, tal vez no necesiten un Sistema Experto.

- + Separación del conocimiento que pertenece a la tarea.  
 ¿Este problema puede ser separado de las otras actividades de los expertos, o esta fuertemente atada a otros tipos de conocimiento y resolución de problemas?  
 ¿Se puede identificar los tipos de conocimiento dependientes de los problemas y las estrategias que están utilizando los expertos, o estos se fían de su sentido común y conocimiento genérico?.

Nuevamente al entrevistar a una variedad de expertos se puede realizar un mejor juicio al respecto.

- + ¿Que conocimiento será capturado en el sistema?  
 ¿Existe algo en común en como resuelven los problemas los expertos, o cada uno tiene diferente aproximación según su idiosincrasia?  
 ¿Existen diferencias en la aproximación de diferentes subproblemas entre los diferentes expertos debido a su especialización?

¿La expertez para la resolución de problemas seleccionados reside en ciertos individuos, o involucra a la comunidad?

Estos son asuntos de gran importancia en la naturaleza de la expertez necesaria en un sistema experto. Es necesario tomar diferentes medidas para resolver estos asuntos. La medida típica esta basada en la identificación de expertos claves en su campo y en basar el sistema en su conocimiento y experiencia.

Se necesita mucha investigación para resolver los diferentes aspectos de un problema, y en lo que respecta a los expertos, es necesario integrar el conocimiento de varios de ellos, resolver conflictos, personalizar bases de conocimiento de la comunidad y desarrollar técnicas de programación para soportar estas actividades.



**IV. MALLA DE ADQUISICIÓN DEL CONOCIMIENTO.**

El método de Malla de Adquisición del Conocimiento fue desarrollado para ayudar al Ing. del Conocimiento en la transferencia manual de la expertez. Esta Malla describe un espacio bidimensional en el cual cinco formas de conocimiento experto constituyen la dimensión horizontal y seis tipos básicos de preguntas de entrevista la vertical.

Este método fue desarrollado para combatir algunos de los elementos que provocan que la Adquisición del Conocimiento sea un cuello de botella durante el desarrollo de un Sistema Experto. Algunos de estos elementos son:

- 1) falta de conocimiento de la naturaleza de la expertez,
- 2) falta de habilidades para la entrevista manual y
- 3) un repertorio limitado de estrategias de preguntas entre muchos otros.

A continuación se describen brevemente estos elementos y la solución que plantea este método:

**Naturaleza de la expertez.**

Los Ing. del Conocimiento que asumen que el problema de la Adquisición del Conocimiento es esencialmente cuantitativo, podrían ignorar conocimiento y estrategias expertas importantes. La psicología cognitiva ha mostrado que los expertos difieren de los novatos no solo en la cantidad de conocimiento que poseen sino en la forma en que este conocimiento es representado y relacionado. (Chi, Glaser & Rees 1982, Larkin, McDermott, Simon & Simon 1980, Milojkovic 1982).

La concepción del conocimiento experto desde el punto de vista cualitativo tiene implicaciones importantes para la gente involucrada con la Adquisición del Conocimiento de los expertos. La dimensión de las Formas de Conocimiento de la Malla de Adquisición del Conocimiento enfrenta este elemento y articula cinco tipos diferentes de conocimiento experto, cada uno con su propia estructura y características e incluyendo un subconjunto de expertez del experto.

**Pocas habilidades para la entrevista manual.**

El proceso de la Adquisición del Conocimiento a menudo demanda una intensa colaboración entre Ing. del Conocimiento y el experto del dominio. Esto como ya se mencionó requiere competencia interpersonal y habilidades de comunicación. Tanto los Ing. del Conocimiento como los expertos tienen diferentes experiencias y metas así como sus maneras de pensar y resolver los problemas. Si estas dificultades son ignoradas, el entendimiento que tenga el Ing. del Conocimiento sobre un dominio particular puede ser incompleto o estar erróneamente construido.

A los expertos no se les puede tratar como objetos que pueden ser simplemente conectados y drenados de todo lo que saben. Una manera de solucionar este problema es tomar en cuenta la naturaleza psicológica de la expertez y las cualidades interpersonales de adquisición y representación.

**Conjunto limitado de estrategias de preguntas.**

El contar con un proceso de Adquisición del Conocimiento más eficiente, depende de la cantidad de maneras de llegar a la expertez humana. No es suficiente el tener la paciencia de preguntarle al experto 400 veces "por que" en un renglón, por el contrario se requiere un repertorio amplio y encausado de preguntas y pruebas. Algunas investigaciones en psicología cognitiva han mostrado que diferentes tipos de preguntas producen diferentes tipos de repuestas (Clark 1985, Loftus 1975). La Dimensión de Tipos de Preguntas en la Malla de Adquisición del Conocimiento articula seis tipos distintos de preguntas dirigidas a diferentes aspectos del conocimiento experto.

Estos elementos que generan el cuello de botella de la Adquisición del Conocimiento pueden ser substancialmente reducidos utilizando un método manual que comprenda la naturaleza del proceso de adquisición y sea adecuado para comprender diferentes tipos de expertez utilizando un conjunto diverso de preguntas.

**VI.I Estructura de la Malla de Adquisición del Conocimiento.**

La Malla de Adquisición del Conocimiento organiza el conocimiento experto y las preguntas del Ing. del Conocimiento en dimensiones separadas pero manteniéndolas en interacción. La primera dimensión, representada por los renglones en la Malla, describe las Formas del Conocimiento en las que la expertez y estrategias del experto son almacenadas. La segunda dimensión representada por los renglones en la Malla, describe los Tipos de Preguntas que están disponibles para los Ing. del Conocimiento con la finalidad de hacerlas explícitas.

**DIMENSION 1. FORMAS DEL CONOCIMIENTO.**

Como ya se mencionó, la representación interna del conocimiento que posee el experto es complicada, por lo que los encargados de construir un sistema experto necesitan una manera de conocer y acceder el material que está distribuido en la memoria y hacerlo explícito. La dimensión de las Formas del conocimiento describe cinco categorías distintas. A continuación se da una breve descripción de cada una de ellas:

**Presentaciones.**

Las presentaciones incorporan el "mapa" de la tarea que posee el experto, incluyendo un entendimiento de sus alcances, organización y clasificaciones básicas. Las presentaciones categorizan y dan coherencia a los hechos y heurísticas del experto al especificar las metas que desea alcanzar y los criterios utilizados para caracterizar un problema dado. Al tener acceso a la manera en la cual el experto toma la tarea y organiza la información actual en base al conocimiento anterior y al contexto presente, el Ing. del Conocimiento está mejor capacitado para ubicar el problema dentro del marco de trabajo.

**Historias.**

Las historias representan los casos clásicos y los ejemplos típicos escogidos de la enorme experiencia del experto con el problema del dominio. Por ejemplo las Historias Explicatorias son aquellas en las cuales el experto trata de dar explicación a situaciones enredosas y narra un conjunto de eventos que expone de tal manera que puedan conducir y producir el fenómeno en cuestión. Una historia de Diagnóstico o Prescriptiva describe algunos fenómenos de tal manera que muestra que es lo que está mal y qué necesita ser arreglado.

**Guiones.**

Los guiones proporcionan el conocimiento secuencial y procedural del dominio. Los elementos básicos de los guiones son los roles, los apoyos estándares u objetos necesarios para llevar a cabo las acciones, una secuencia estandar de escenas en las cuales un acto habilita el siguiente y algunos resultados normales de la ejecución exitosa de alguna actividad. Para conocer el guión de un experto, se debe tener un gráfica temporal de acciones críticas y estar capacitado para entender cada acción en términos del conocimiento requerido apriori para realizar tal acción.

**Metáforas.**

Las metáforas encapsulan las imágenes alternativas que posee el experto sobre la tarea, cada una de las cuales incluye características únicas, limitantes y opciones. Las metáforas describen un elemento al hacer referencia a otro aparentemente disimilar de manera tal que el primero es entendido de manera más completa que si no se hubiese realizado la comparación. La ventaja de la metáfora es que está capacitada para presentar una idea que posteriormente puede ser reconstruída y mejorada a través de pruebas directas del Ing. del Conocimiento al experto del dominio.

**Reglas de Dedc.**

Las reglas de dedo proveen un conjunto innumerable de heurísticas para interpretar y trabajar con un arreglo de circunstancias encontradas al realizar la tarea. Una regla de dedo encapsula conocimiento tácito acerca de las condiciones que garantizan alguna acción dada y acerca de cómo recolectar datos y evaluar las condiciones actuales. Las reglas de dedo son estrategias concretas e implementables de un alcance de menor a moderado, las cuales pueden separar y definir como asuntos aquellas condiciones específicas y limitadas para las cuales sirven como la estrategia más completa.

Cada forma de conocimiento es una manera diferente en la cual el conocimiento puede ser representado en la mente del experto. Las cinco formas de conocimiento juntas sugieren que la expertez en un dominio dado no está codificada en la mente del experto como un solo conjunto, sino que la variedad de experiencias con el problema producen diferentes representaciones mentales.

El soporte para estas ideas proviene de las investigaciones en psicología cognitiva que muestra que el aprendizaje es un proceso segmentado y que los diferentes segmentos son almacenados en diferentes partes de la memoria (Anderson 1980). Los psicólogos han realizado hipótesis que particionan la información de tal manera que la gente hace mejores generalizaciones y más útiles predicciones. De manera similar, la teoría de la "Sociedad de la Mente" de Minsky propone que las acciones inteligentes emergen de las interacciones de muchos sistemas pequeños operando dentro de una estructura administrativa en evolución (Minsky 1986).

La dimensión de las Formas del Conocimiento ve la expertez como algo construido a partir de un número de subsistemas pequeños, cada uno con su propio carácter y función. Adicionalmente, existe un número de ventajas prácticas de ver la expertez como multi-modo. En primer lugar las Formas del Conocimiento proveen una manera, durante la Adquisición del Conocimiento, de catalogar la complejidad real del dominio. Los cinco tipos de conocimiento proveen el significado para manipular los aspectos complejos del dominio. En segundo lugar las formas del conocimiento estimulan una cobertura amplia al ver el conocimiento como multi-modo, de esta manera el Ing. del Conocimiento está mejor capacitado para visualizar cuando los otros modos de expresar la expertez se volverán relevantes. En tercer lugar cada forma de conocimiento refleja un punto de vista del problema diferente pero práctico. Las presentaciones, historias, guiones, metáforas y reglas de dedo describen de manera implícita cómo enfrentar las diferentes características de la tarea.

Las cinco Formas del Conocimiento tienen la ventaja de permitir un entendimiento del dominio del experto en sus propias palabras. Esto proporciona al Ing. del Conocimiento una manera de acercarse a los conceptos del dominio y al mismo tiempo tener los conocimientos para organizarlos. Finalmente las cinco Formas del Conocimiento proveen una conexión entre la Adquisición del Conocimiento del experto y la transformación en una representación en particular. Esto permite que la expertez primero sea entendida en sus propios términos lo cual es muy importante, para posteriormente ser capaces de seleccionar la representación adecuada para ellos.

Existe una advertencia: aunque el conocimiento experto generalmente es multi-modo, no todas las Formas del Conocimiento son evidentes en todos los dominios del conocimiento, ni son utilizados de manera equivalente por todos los expertos. No obstante los Ing. del Conocimiento deben reconocer el potencial de cada área y no asumir que las formas no familiares son necesariamente las inproductivas.

**DIMENSION 2: TIPOS DE PREGUNTAS.**

El proceso de construcción de una Base de Conocimientos requiere que el Ing. del Conocimiento tenga a mano un número variado de tipos de preguntas. Las investigaciones han mostrado que la utilización de diferentes formatos de preguntas optimiza la información que se recibirá. Alternativamente, el uso de un solo tipo de preguntas limita fuertemente el tipo de respuestas que pueden ser obtenidas. La Malla de Adquisición del Conocimiento presenta 5 tipos de preguntas. A continuación se da una breve descripción de cada tipo junto con un ejemplo.

**Preguntas Grand Tour.**

Las preguntas Grand Tour arrojan una red amplia sobre el dominio con el objeto de detectar los límites que el experto determina en él. El material buscado incluye una revisión de las perspectivas, metas, organización y clasificaciones del experto. Un ejemplo de una pregunta Grand Tour es: "Podría ud. describir el tipo de actividades que realizan los planificadores?. Por favor no omita ninguna actividad de su descripción, aunque ud. piense que no es importante". Este tipo particular de pregunta está diseñada para adquirir del experto una descripción de tipo Presentación, aunque de igual manera podría estar dirigida hacia adquisición de una Historia o una Metáfora.

**Catalogación de las Categorías.**

El resultado esperado de este tipo de preguntas es un clasificación organizada de los términos y conceptos del experto. Un ejemplo de una pregunta de este tipo es: "Cuando ud. me dio una perspectiva de su trabajo, ud. habló sobre los planificadores. Existen diferente tipos de planificadores?. Los planificadores son un subtipo de algún otro tipo de trabajo?". Este tipo de preguntas podría producir una respuesta de tipo Presentación o una pregunta Grand Tour.

**Investigación de Atributos.**

Esta preguntas impulsan a descubrir las características sobresalientes y un rango de posibles valores de los conceptos del experto. Un ejemplo de una pregunta de investigación de atributos es: "Ud. ha descrito un número de tipos de situaciones del plan que ud. ha encontrado. Me

pregunto si ud, podría tomar las dos primeras que mencionó y describir algunas maneras en las que se asemejan pero que las hacen diferentes al tercer ejemplo que usted dio?". Este tipo particular de pregunta persigue una descripción de tipo Historia.

#### **Determinación de Interconexiones.**

Estas preguntas están dirigidas a descubrir las relaciones entre los conceptos en el dominio. La existencia de un modelo causal para todo el dominio o parte de él, es de particular interés. Un ejemplo de una pregunta de determinación de interconexiones es: "En la descripción del conjunto de pasos de rutina para planificar una orden, usted dijo que la revisión de la fecha requerida ocurre antes que cualquier otra cosa. Porque sucede esto?". esta pregunta está dirigida a obtener información de tipo Guión.

#### **Buscando Consejo.**

Este tipo de pregunta está diseñado para obtener recomendaciones expertas y por lo tanto estrategias para enfrentar una variedad de condiciones tales como, la manera de determinar las condiciones actuales y que condiciones garantizan una acción dada. Un ejemplo de este tipo de pregunta es: "Usted ha comparado la planificación con un juego de mesa; de su experiencia con juegos de mesa que consejo podría dar cuando se planifica una orden?". Este tipo de preguntas podría generar una respuesta de tipo Metáfora.

#### **Preguntas de chequeo cruzado.**

Este tipo de preguntas está diseñado para validar y examinar los límites de la información previamente obtenida. Este tipo de preguntas actualmente consiste de cinco subtipos incluyendo a Preguntas Ingenuas, Jugando Defensor del Demonio, Formulación de situaciones Hipotéticas, Preguntando ¿Que tan seguro está usted?, y Buscando la excepción. Un ejemplo de una Pregunta Ingenua es "Me podría decir porque las ordenes necesitan ser planificadas?. Un ejemplo de una pregunta de tipo Defensor del Demonio es "Usted dijo que era necesario establecer prioridades en caso de que se planificaran ordenes múltiples. Que sucede si no se establecen tales prioridades?".

Las investigaciones indican que las preguntas restringen y dirigen los tipos de respuestas que pueden ser obtenidas. Por ejemplo existen dos tipos principales de preguntas: las preguntas que piden al receptor que proporcione el elemento requerido, y segundo, expresiones que requieren que el oyente únicamente este de acuerdo o en desacuerdo con el contenido de la pregunta.

De igual manera se ha demostrado que las variaciones sutiles en como se arma la pregunta influye en la respuesta del oyente. Por ejemplo, las investigaciones sobre los testimonios de testigos visuales revelan que los testigos de un accidente automovilístico reportan que el vehículo estaba yendo muy rápido si la pregunta es expresada así: "Que tan rápido estaba yendo el vehículo cuando se estrelló contra la pared?" que si fuese expresada así: "Que tan rápido iba el vehículo cuando corría hacia la pared?".

Otras investigaciones muestran que algunas preguntas obtiene algunos elementos que son ocultados cuando se realizan otro tipo de preguntas. Un segundo resultado de esta investigación reporta que existen varios caminos hacia el evento codificado, de manera que la información que no es accesible por un camino, puede serlo por otro. (Tulving 1974).

La malla de Adquisición del Conocimiento está basada en este marco de trabajo teórico. La matriz formada por las Formas de Conocimiento y los Tipos de Preguntas está diseñada para incrementar el traslape entre la codificación de la expertez por el experto y las subsecuentes recuperaciones de tal expertez por el Ing. del Conocimiento en un contexto de Adquisición del Conocimiento.

Existen muchas ventajas en tener a la mano varios tipos de preguntas. Los seis tipos de preguntas en la malla están diseñados para verse como un paquete cuyo contenido total es adaptado para producir una inspección del conocimiento del experto. Una segunda ventaja es que las preguntas permiten al Ing. del Conocimiento a acercarse al dominio desde diferentes direcciones, de manera tal que incrementa sus oportunidades de que le sea revelado material importante que pudo estar oculto desde otro tipo de direcciones. Una tercera ventaja de contar con diferentes tipos de preguntas es que cada una enfoca su atención hacia diferentes características. Cada pregunta está dirigida a descubrir diferentes aspectos de la expertez.



Los seis tipos de preguntas también varían en el nivel de especificación, que en combinación tienen la ventaja de poder obtener material con diferente detalle y composición. Las preguntas de Grand Tour buscan esbozar descripciones amplias y comprensivas con el objetivo de comprender el alcance del dominio del problema, mientras que las preguntas de chequeo cruzado están dirigidas a verificar detalles finos. Una quinta ventaja de utilizar diferentes tipos de preguntas es el inhibir la influencia del Ing. del Conocimiento. El rango de preguntas debe eliminar la tendencia de los Ing. del Conocimiento de favorecer un tipo de pregunta sobre otro.

La sexta es que al tener acceso a un número de tipos de preguntas diferentes, se protege contra las suposiciones del experto así como del Ing. del Conocimiento. Al realizar un tipo de pregunta diferente sobre el mismo material, el Ing. del Conocimiento puede darse cuenta de que el experto asume incorrectamente ciertas conexiones o inclusiones. Adicionalmente, las preguntas alternativas pueden animar al experto a ser más explícito acerca de algunas cosas que para él solían ser obvias. Finalmente, el tener acceso a diferentes tipos de preguntas permite al Ing. del Conocimiento cambiar la dirección o el contenido y de esta manera mantener el interés y motivación del experto.

#### IV.II Estrategia para el uso de la malla de Adquisición del Conocimiento.

Las cinco Formas de Conocimiento y los seis Tipos de Preguntas en la malla no agota ninguna dirección. En su lugar cada dimensión tiene el objetivo de proveer al Ing. del Conocimiento con un elemento que le permita ver mas expertez relevante. Por otra parte, las categorías dentro de cada dimensión se traslapan en algún grado. Por ejemplo una Historia sobre una situación en particular puede contener una Metáfora o una Regla de dedo. La característica importante de la naturaleza de la matriz de la Malla no es su independencia de las categorías, sino su alcance del dominio.

La malla puede ser utilizada ya sea pasivamente ayudando en la decodificación de las respuestas del experto, o en la obtención del contenido de un conocimiento en particular. A continuación se detallan estos aspectos.

**Decodificación de las respuestas del experto.**

La malla proporciona una descripción explícita de las relaciones recíprocas que pueden existir entre las preguntas de la Adquisición del Conocimiento y las descripciones del experto. La Malla provee una manera de afrontar el tipo de respuestas que no parece estar vinculada a la pregunta que se realizó. La Malla proporciona una herramienta para apreciar e identificar las diferentes formas en las que viene la expertez.

Una de las señales de un buen Ing. del Conocimiento es su habilidad para decodificar respuestas de todos los tipos y especialmente aquellas que no parecen ser una respuesta a la pregunta precedente. Los Ing. del Conocimiento que utilizan la Malla, desarrollan el hábito de preguntarse a sí mismos después de cada respuesta: "Para qué pregunta esta es una respuesta?" y "De qué Forma de conocimiento este es un ejemplo?".

**Evocando Formas particulares de Conocimiento.**

La Malla proporciona el significado para evocar deliberadamente un tipo particular de respuesta del experto. En ocasiones al experto se le puede pedir una metáfora para abrir una nueva línea de investigación en el dominio del problema. O se puede solicitar una historia sobre una situación atípica para obtener un punto de vista diferente después de haber analizado un caso típico.

Dado que frecuentemente los expertos son fuertemente presionados a describir lo que ellos saben, a menudo las preguntas "como" y "porque" no obtienen información importante, pero sin embargo pueden proporcionar un conexión mental como respuesta a un tipo particular de experiencia con el dominio. De esta manera el material previamente implícito puede hacerse explícito.

**Caminos Alternativos a través de la Malla.**

Cada dimensión de la Malla tiene su propio orden interno. En la dimensión de Tipos de Preguntas, las preguntas abiertas están en la parte superior, mientras que a medida que se hacen explícitas, las preguntas se van colocando hacia la parte inferior. La dimensión de las Formas del Conocimiento es menos lineal. Los tipos de conocimiento más declarativo son representados hacia la izquierda y los tipos de conocimiento más procedural están hacia el lado derecho de la malla.

Una manera de utilizar la Malla es comenzar en la parte superior izquierda y moverse hacia atrás y adelante en una manera más o menos sistemática hacia la parte inferior derecha. De esta manera comienza con una amplia investigación de los aspectos basados en hechos de la expertez y concluye con la atención enfocada en heurísticas específicas.

Sin embargo esta táctica no es la única manera de utilizar la Malla. También puede ser utilizada de una manera cíclica. El Ing. del Conocimiento se mueve de las preguntas generales a las específicas y regresa a las generales y del conocimiento declarativo al procedural y de regreso al declarativo.

El como se utilice la Malla finalmente dependerá de las necesidades particulares de los diferentes equipos de Sistemas Expertos. El valor de la Malla consiste de su descripción multi-modo del conocimiento experto y del significado multi-forma de la investigación realizada para capturar el conocimiento.

## V. CASOS O EJEMPLOS.

Este método estructura la adquisición del conocimiento alrededor de casos. Un caso es un ejemplo ya trabajado, que identifica el problema en cuestión (ej: el agua de las hieleras no se congeló) y enseña cada elemento de información adquirido por el experto para realizar su análisis (ej: la luz del refrigerador está apagada cuando se abre la puerta de este). Para cada elemento identifica por qué la información fue buscada (ej: Si la luz está pagada, probablemente el refrigerador no tiene electricidad).

Cada caso puede servir como una base sobre la cual se puede incrementar el conocimiento que se tiene codificado en el sistema. A partir de esta base es posible realizar una exploración sistemática para identificar cuál de los diferentes caminos seguiría el análisis y así obtener los requerimientos de información que produce cada resultado (ej: supongamos que se reviso el cordón de electricidad y esta enchufado. Que se haría?. Se debería enchufar otro aparato para ver si la toma de corriente funciona o no).

Si se procede de esta manera, es posible desarrollar una estructura jerárquica del conocimiento descrito durante el proceso. Esta estructura es de gran utilidad en el sistema pues puede ser utilizada como una librería de preguntas cuando se cuestione al sistema sobre el por qué de una decisión o el por qué de una respuesta dada.

A medida que se vaya haciendo cambios significativos al sistema, se pueden recorrer todos los casos previamente trabajados para asegurarse que el sistema todavía los maneja apropiada y correctamente.

## VI. INDUCCIÓN.

Este método sirve tanto para la Adquisición del Conocimiento como para el aprendizaje de la máquina.

La inducción es lo contrario a la deducción. En la deducción se da una regla general de la cual es posible deducir hechos sobre casos específicos. La inducción trabaja de una manera inversa es decir, dado un conjunto de ejemplos específicos, estos son investigados y se inducen reglas o patrones. Las reglas inducidas aplican al conjunto de ejemplos, y se espera que apliquen a cualquier ejemplo en general. Sin embargo no se puede estar seguro de que los resultados sean correctos. Por otra parte la calidad de los resultados inducidos, dependerá tanto del algoritmo como del conjunto de ejemplos que se tenga disponible.

Este método es bueno pues es común que el experto sea subjetivo y olvidadizo, que omita detalles o que pueda ser inconsistente, por lo que es más fácil para él, referirse a ejemplos específicos que describir sus procesos. Si el experto puede aportar un conjunto de ejemplos de casos, consistentes con sus decisiones, junto con los factores (llamados atributos) que el tome en cuenta en la toma de decisiones, entonces un programa podrá inducir reglas a partir de tales ejemplos. Estos ejemplos son llamados "conjunto de formación". Los resultados muestran que es posible obtener un mejor desempeño con este método que entrevistando al experto[3].

La inducción tiene ciertas ventajas, ya que es objetiva, repetible, consistente y la salida es fácil de entender.

La parte clave en la inducción es la selección del conjunto de formación. Ningún algoritmo puede inducir algo que no existe. La selección de atributos no es un procedimiento al azar y demanda la habilidad del experto. Un conjunto de formación pobre de atributos o mal elegido producirá un conjunto pobre de reglas.

En este método, no solo es importante la selección de atributos, sino también la selección de los ejemplos. Las reglas inducidas deben reflejar tanto a casos raros y difíciles como a los comunes y fáciles. Esto significa que el

conjunto de formación, idealmente, debe comprender un ejemplo de cada tipo diferente (en términos de procesos de decisión) que necesite ser incluido en las reglas. El problema que se presenta en este punto es que debido a que las reglas son desconocidas, es difícil juzgar la calidad de los ejemplos.

Una vez que el conjunto de formación está disponible, la inducción puede ser utilizada. La inducción debe ser vista como un método de obtención de preguntas así como de sus respuestas. Esto puede identificar contradicciones, errores, casos interesantes o atributos importantes.

## VII. TÉCNICA DE LA MALLA DE REPERTORIO

G.A. Kelly desarrolló una teoría de construcción personal, en la que afirma que cada persona tiene su propio modelo del mundo, creada a partir de sus construcciones o experiencias personales.

La malla de Repertorio es un método de investigación semejante a un modelo. El modelo consiste de elementos y constructores. Los constructores son análogos a los atributos mencionados en el método de inducción con la diferencia de que los constructores deben ser bipolares o medibles por ejemplo : debil/fuerte, verdadero/falso, tamaño, peso. "Color" por ejemplo no sería un constructor, sin embargo grado de coloración si podría serlo. Los constructores son la manera en la que pares de elementos pueden ser descritos como similares o diferentes por ejemplo A es fuerte pero B es débil; C y D son ambos verdaderos, etc. Los elementos son análogos a los ejemplos en el método de inducción y son elegidos por el usuario como los elementos que son importantes para él, por lo que no puede existir una elección mala.

Para comenzar es indispensable definir un problema en particular para que el experto piense en él, de manera que posteriormente defina los elementos y constructores que considera relevantes para este problema en particular.

La "malla de repertorio" es un sistema de referencias cruzadas entre los constructores y los elementos para tal problema. Esta es sucesivamente refinada hasta que el usuario quede satisfecho con el resultado. De esta manera se obliga al experto a investigar cómo piensa acerca del problema. Por ejemplo el problema puede ser "Que hace que un diseño sea bueno", donde los elementos serían ejemplos clave de un buen y un mal diseño, y los constructores serían las características de estos diseños.

Existen varias maneras mediante las cuales se puede obtener las mallas. El experto elige los elementos que considera que son los más relevantes del problema en discusión. Los constructores pueden ser proporcionados de manera similar por el experto o mediante un análisis sistemático de los elementos. Un método posible es aquel que selecciona grupos de tres elementos al azar y pide al experto que los divida en dos similares y uno distinto, y así obtiene el constructor

que describe esta distinción. Habiendo obtenido los constructores, el experto califica cada elemento de acuerdo a su constructor. Esta calificación puede tener un valor de verdadero o falso o un valor subjetivo en una escala numérica de 1 a n. En cualquier estado de la adquisición el experto puede adicionar mas elementos o constructores o alterar las entradas de la malla. De esta manera, el proceso aumenta su conocimiento de como el experto ve el problema.

El proceso mediante el cual el experto re-analiza la malla es invaluable y podría ser una técnica útil para la extracción de atributos para la inducción. Sin embargo, también es posible analizar la malla para identificar patrones. Un método útil es usar análisis de grupo para identificar las agrupaciones que son similares o correlativas y también los elementos que son similares. Habiendo obtenido estos resultados, la malla puede ser re-ordenada o enfocada para representar un modelo coherente de la visión del experto. Si el experto no está de acuerdo con los resultados, entonces puede modificar la malla hasta que esta represente mejor su percepción de los usos.

**Ejemplo de una malla que muestra la evaluación de la programación.**

La Fig. 3.2 muestra los resultados de una consulta hecha a un lider de proyecto. Este evaluó las habilidades de programación de sus programadores. Antes de la consulta, el lider era capaz de describir "buenos programas" únicamente en términos abstractos como legible, mantenible y claridad, términos que a su vez no pudo definir fácilmente.

Se le pidió nombrar programadores específicos y las características de sus programas. A medida que nombraba a sus programadores, evaluaba su trabajo. La Fig. 3.1 muestra los resultados de su investigación. El lider tenía los elementos suficientes para sumarizar los programas y los suficientes constructores para hacer una distinción entre ellos. Este quedó satisfecho con las característica principales que había descrito de los programas de manera que no fue necesaria ninguna alteración a la malla.



Investigación de programas de computadoras										
		E1	E2	E3	E4	E5	E6	E7		
C1	Variables pobres	4	2	3	5	3	1	1	Nombres significativos	C1
C2	Sin comentarios	5	2	5	4	4	3	1	Comentarios	C2
C3	Estructura Pobre	5	2	1	4	5	2	2	Estructurado	C3
C4	Mala presentación	5	2	4	3	4	1	2	Buena presentación	C4
C5	Sin Go To	1	4	5	1	1	5	5	Go To's	C5
C6	Ilegible	5	1	3	4	4	1	2	Legible	C6
C7	Sin procedimientos	5	1	4	5	5	5	1	Modular	C7

Ana	_____	_____	_____	_____	_____	_____	_____	_____
Concepcion	_____	_____	_____	_____	_____	_____	_____	_____
Alfredo	_____	_____	_____	_____	_____	_____	_____	_____
Edgar	_____	_____	_____	_____	_____	_____	_____	_____
Miguel	_____	_____	_____	_____	_____	_____	_____	_____
Patricia	_____	_____	_____	_____	_____	_____	_____	_____
Arturo	_____	_____	_____	_____	_____	_____	_____	_____

Fig 3.2 Malla adquirida para la evaluación de técnicas de programación [3]

En la malla del presente ejemplo, los elementos E1,...,E7 representan a los programadores y los constructores C1,...,C7 representan las características elegidas por el líder de proyecto para calificar los programas. Los ratings asignados van de 1 a 5 siendo 5 la mayor o mejor calificación.

Los resultados que se muestran en la fig. 3.3 se obtuvieron mediante un proceso que es una interpretación personal de los pasos que se deben realizar para analizar la malla. Este proceso describe a continuación.

		E7	E2	E5	E3	E4	E5	E1	
C7	Sin procedimientos	1	1	5	4	5	5	5	Modular
C2	Sin comentarios	1	2	3	5	4	4	5	Comentarios
C4	Mala presentación	2	2	1	4	3	4	5	Buena presentación
C5'	Go To 's	1	2	1	1	5	5	5	Sin Go To's
C1	Nombres pobres	1	2	1	3	5	3	4	Nombres significativos
C3	Estructura Pobre	2	2	2	1	4	5	5	Estructurado
C6	Ilegible	2	1	1	3	4	4	5	Legible

Mal	_____	_____	_____	_____	_____	_____	_____	_____	
Muy Pobre	_____	_____	_____	_____	_____	_____	_____	_____	
Pobre	_____	_____	_____	_____	_____	_____	_____	_____	
Regular	_____	_____	_____	_____	_____	_____	_____	_____	
Bueno	_____	_____	_____	_____	_____	_____	_____	_____	
Bueno	_____	_____	_____	_____	_____	_____	_____	_____	
Muy Bueno	_____	_____	_____	_____	_____	_____	_____	_____	

Fig 3.3 Malla enfocada para la evaluación de técnicas de programación [3]

	E1	E2	E3	E4	E5	E6	E7
E1		22	13	6	4	28	24
E2	22		13	18	18	8	4
E3	13	13		13	11	11	13
E4	6	18	13		4	16	28
E5	4	18	11	4		16	28
E6	28	8	11	16	16		18
E7	24	4	13	28	28	18	

Fig 3.4 Tabla de Evaluación de columnas

En principio, es necesario estandarizar las características, es decir, si existe alguna que sea negativa o no deseable, es necesario complementarla así como sus ratings, por ejemplo, los go to's son una característica no deseable, por lo cual debe ser complementada a NO go to's y sus ratings deben ser restados a 6 de manera que la mayor calificación: 5, será complementada a la menor :1.

Posteriormente se debe hallar la medida de la diferencia entre dos ejemplos mediante la suma de los valores absolutos de la diferencia de los ratings. Esto permite que los elementos similares sean puestos juntos en la malla. De esta manera se obtiene la tabla que se muestra en la fig 3.4 de donde se puede apreciar que el elemento mas cercano

a E1 es E5,	a E2 es E7,	a E3 es E5 y E6
a E4 es E5,	a E5 es E1 y E4	a E6 es E7
a E7 es E2		

por lo que los elementos son reorganizados de la siguiente manera

E7 -> E2 -> E6 -> E3 -> E4 -> E5 -> E1

Adicionalmente se obtuvieron los ratings totales de cada ejemplo o programador y fueron evaluados como se puede apreciar a continuación:

E1 = 34	Muy Bueno
E2 = 12	Muy Pobre
E3 = 21	Regular
E4 = 30	Bueno
E5 = 30	Bueno
E6 = 14	Pobre
E7 = 10	Mal

Para reorganizar los renglones se deben seguir los siguientes criterios para seleccionar sucesivamente cada uno:

- Se selecciona el renglón que tenga el mayor rating mas a la izquierda.

- En caso de que exista el mismo rating en mas de un renglón, se deberá tomar el que tenga el mayor rating en la columna de su izquierda.
- Si nuevamente existe mas de un renglón con el mismo rating, se debe seleccionar aquel renglón que tenga los mayores ratings en las columnas de su derecha.

Siguiendo los criterios antes mencionados se obtuvo la malla que se muestra en la fig 3.3, la salida presenta una representación de la opinión del experto. El experto puede hallarlo útil para clasificar sus ideas y se obtuvieron las siguientes observaciones basadas en la malla.

- La instrucción GO TO y la buena estructura, no van juntos.
- Una buena presentación probablemente significa que el programa es legible.
- Si alguien utiliza comentarios, es muy probable que utilice nombres significativos.
- La modularidad es un asunto diferente.
- Es más fácil definir muy buenos o muy malos programas.
- Es necesario animar a los programadores a adquirir las características que les hacen falta.
- Es más claro definir "estructura, legibilidad, modularidad, etc."

Este problema muestra como un problema pobremente definido puede clarificarse utilizando este método. La inducción habría sido de poca ayuda en este estado por haber muchas clasificaciones y el experto se hubiera confundido acerca de las relaciones entre los constructores. Este método puede identificar las correlaciones entre los constructores, en la inducción los atributos correlacionados deben ser tratados con cuidado. Existen técnicas para comparar dos mallas de dos personas diferentes para investigar como difieren sus puntos de vista. Este método no proporciona reglas pero identifica conceptos similares y es muy útil en la codificación de conocimiento.

La aplicación interactiva de la técnica de Malla de Repertorio para el desarrollo de Sistemas Expertos, fue propuesta por primera vez en 1980 por Gaines y Shaw. Posteriormente, en 1984, Boose demostró su efectividad en aplicaciones industriales y actualmente a tenido un uso creciente en el desarrollo de Sistemas Expertos. La Fig 3-5 muestra un "árbol genealógico" de los desarrollos más sobresalientes de herramientas de Adquisición del Conocimiento que utilizan la técnica de malla de repertorio y la extensión del concepto original de Kelly [26].

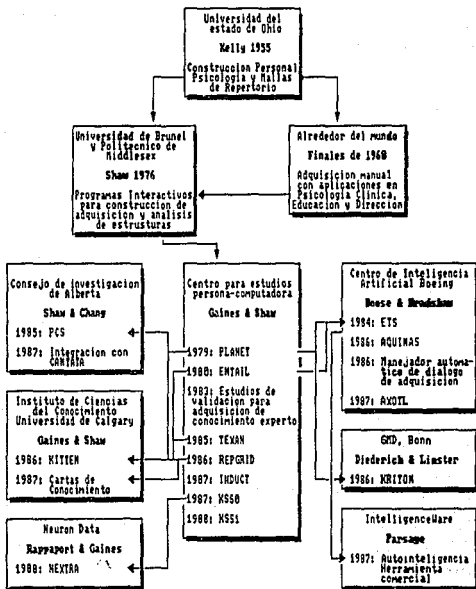


Fig 3.5 "Árbol Genealógico" de Aplicaciones y desarrollo de la Malla de Repertorio [26]

#### CAPTITULO 4

### TECNICAS DE ADOQUISICION DEL CONOCIMIENTO:

#### ENFOQUE AUTOMATIZADO

En los últimos años, el esfuerzo por perfeccionar las investigaciones sobre el proceso de la Adquisición del Conocimiento para Sistemas Expertos, ha cada vez mayor. Este esfuerzo difiere según el grado en el que se apegue al paradigma tradicional de: Experto -> Ing. del Conocimiento -> Base de Conocimientos. La mayoría de los esfuerzos pretenden reducir la dependencia del Ing. del Conocimiento a través del desarrollo de sistemas inteligentes de edición que faciliten la edición de la Base de Conocimientos, de manera que se logre alcanzar un enfoque del tipo: Experto -> Sistema de Adquisición de Conocimientos -> Base de Conocimientos.

Cualquier sistema que sea desarrollado con la intención de automatizar el proceso de Adquisición debe realizar el mismo tipo de trabajo que realiza el Ing. del Conocimiento. Esto significa que tal sistema debe conocer:

- Conocer el vocabulario básico del dominio,
- identificarlas piezas faltantes de información,
- integrar la información nueva en Bases de Conocimientos parcialmente desarrolladas ,
- y adquirir explicaciones y aclaraciones cuando la información del experto este incompleta o inconsistente,
- Finalmente la Base de Conocimientos debe ser organizada con el objetivo de maximizar la transparencia para los usuarios finales.

Para realizar tal trabajo, el sistema de Adquisición requiere modelar el dominio y transportar tal modelo al proceso de adquisición.

En el presente capítulo se desarrollan dos técnicas que se fundamentan en un enfoque diferente de modelado del dominio, estas técnicas son:

- Adquisición del Conocimiento a través de técnicas de aprendizaje de conceptos, y
- Adquisición del Conocimiento procedural de los expertos del dominio.

## I. ADQUISICIÓN DEL CONOCIMIENTO A TRAVÉS DE TÉCNICAS DE APRENDIZAJE DE CONCEPTOS.

Mediante el esquema de aprendizaje de conceptos se puede lograr que el experto tenga interacción directa con las máquinas para transferir su conocimiento[4]. Aquí se presenta un marco de trabajo interactivo para la descripción de técnicas de aprendizaje de conceptos. Este provee las bases generales para relacionar el aprendizaje de conceptos con la adquisición de conocimientos.

El conocimiento debe ser representado en forma abstracta en los Sistemas Expertos de manera que pueda ser almacenado y manipulado de manera efectiva. Aunque es difícil para un experto expresar su conocimiento en forma de reglas u otro tipo de abstracciones, como ya se vio le es fácil demostrar su expertez en la realización de situaciones específicas. Los esquemas para el aprendizaje son representaciones abstractas - o conceptos - a partir de ejemplos, lo cual permite que los expertos interactuen directamente con la máquina para transferirle su conocimiento.

En años recientes, los estudios sobre el aprendizaje han logrado concretar algunas técnicas básicas para la generalización de ejemplos concretos a descripciones más abstractas. Estas técnicas incorporan heurística para la generalización de tipos particulares de datos, algoritmos de eliminación de candidatos, métodos para la generación de árboles de decisión y de conjuntos de reglas, propagación hacia atrás de confusiones a través de árboles de explicación, funciones de inducción y síntesis de procedimientos para la ejecución de trazos[4].

A pesar de haberse desarrollado investigaciones y programas al respecto, no ha surgido una metodología coherente para describir y categorizar las técnicas de aprendizaje de manera que las haga accesibles a usuarios potenciales.

El método de adquisición de conocimientos a través del aprendizaje de conceptos, ayuda al usuario a identificar las técnicas más apropiadas para el aprendizaje de conceptos para una aplicación dada.

Desde el punto de vista de la Ingeniería del Conocimiento, los sistemas de aprendizaje de conceptos difieren en su representación de los ejemplos y los conceptos, en la manera de influenciar la búsqueda de un concepto determinado y el modo de interacción con el profesor.

## I. Aprendizaje de Conceptos

Los sistemas actuales de Adquisición de Conocimientos realizan rutinas de limpieza que permiten rotar el aprendizaje de los hechos presentes explícitos y están capacitados para obtener del experto reglas simples basadas en los atributos de casos ejemplo. Sin embargo estos no pueden inferir reglas que operen en representaciones más poderosas como las estructuras anidadas, ni pueden aprender de casos presentados por usuarios no expertos. Los métodos de aprendizaje de conceptos pueden estar capacitados para manejar estos problemas pero aunque su estado aún es primitivo y sugiere únicamente ideas en lugar de algoritmos bien desarrollados que sirvan de herramienta en la Adquisición del Conocimiento. Los sistemas de aprendizaje de conceptos toman ejemplos y crean descripciones generales, que a menudo son expresadas como reglas, que necesita el sistema.

La definición de concepto es vaga, pero tiene la idea abstracta de que es una clase de objeto derivado de instancias específicas u ocurrencias. El aprendizaje igualmente es un término de amplio significado y denota la obtención de conocimiento, habilidad y entendimiento a partir de la instrucción, experiencia o reflexión, es decir la Adquisición del Conocimiento de la gente. A pesar de ello en la Inteligencia Artificial este término está más concretamente definido y denota la adquisición de descripciones estructurales a partir de ejemplos de lo que se está describiendo.

El aprendizaje de conceptos involucra la adquisición de descripciones que hacen que la estructura de generalizaciones sea explícita. Para una persona el aprendizaje no es solo cuestión de adquirir una descripción sino que involucra tomar algo nuevo e integrarlo totalmente con los procesos pensantes existente. Los programas de computadora no son tan hábiles, pero sin embargo estos adquieren descripciones que son explícitas en el sentido que pueden ser comunicados económicamente (por ej. en forma de reglas) y pueden soportar cualquier tipo de razonamiento.

El marco de trabajo desarrollado a continuación [4] (Fig 4.1) está basado en distinciones, las cuales son clasificadas en tres áreas:

- Representación de ejemplos y conceptos.
- Tendencias en la búsqueda de conceptos.
- Interacción con el profesor.



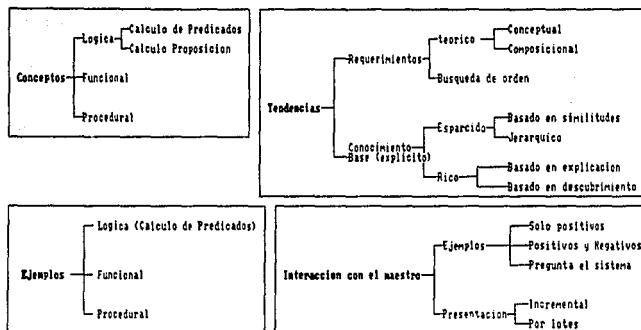


Fig 4.1 Distinciones en el marco de trabajo de aprendizaje de conceptos.

Antes de comenzar con la representación de conceptos y ejemplos es imprescindible mencionar que existen 2 tipos de aprendizaje de conceptos, inductivo y deductivo. Con el primero se debe inferir una conclusión general a partir de ejemplos empíricos, y con el segundo, si se cuenta con una cantidad de ejemplos que son conocidos de manera exhaustiva, el aprendiz podría deducir un concepto.

Los dos tipos de aprendizaje de conceptos involucran una búsqueda, pero mientras los métodos deductivos expresan conceptos en un lenguaje basado en inferencias, los métodos deductivos pueden utilizar descripciones más generales.

Existen dos diferencias básicas entre estos dos métodos:

- En la inducción la búsqueda está gobernada por un ordenamiento complejo, mientras que para la deducción termina cuando una meta predefinida es alcanzada.
- El objeto de interés (el concepto aprendido) proviene de la inducción, mientras que el resultado de la deducción es una generalización del camino mismo de la búsqueda.

## II. Representación de conceptos y ejemplos

Los conceptos, y ejemplos a partir de los cuales son creados, son la salida y entrada de los Sistemas de Adquisición de Conocimientos, es decir lo que es aprendido y lo que es proporcionado por el profesor u otro agente externo.

Para que un marco de trabajo para la representación de conceptos sea útil, debe proveer al Ing. del Conocimiento métodos para la selección apropiada de representaciones de los ejemplos, conceptos y el conocimiento que ya se posee o conocimientos base.

Las representaciones de los ejemplos y los conceptos son requeridas de manera separada pero debe estar relacionada. A continuación se trata cada uno de estos desde tres puntos de vista :

- Lógica
- Funcional y
- Procedural.

### A.- Representación de conceptos.

La distinción más importante en un marco de trabajo para la Adquisición de Conocimientos, es cómo el sistema representa lo que aprende.

La representación de conocimiento ha sido un tópico muy importante en las investigaciones de la Inteligencia Artificial, pero se debe aclarar que una estructura de datos no es conocimiento. La representación de hechos o reglas solo se convierte en conocimiento cuando es utilizado por un programa que se comporta de manera cognoscitiva.

En este marco de trabajo se proponen tres técnicas de aprendizaje de conceptos:

- Cálculo de predicados de primer orden.
- Expresiones compuestas por funciones.
- y Procedimientos.

Estas reflejan fundamentalmente las formulaciones realizadas en el estilo de programación lógica, funcional e imperativa, y aunque son equivalentes en su poder expresivo, sus diferentes representaciones son más o menos apropiadas para algunos problemas específicos de aprendizaje de conceptos, dependiendo de la naturaleza de los ejemplos, del conocimiento que ya se posee, de la manera en la que se mide la complejidad de los conceptos

y del tipo de interacción que se tiene con el maestro, por ejemplo la representación funcional incorpora las matemáticas de manera natural, los procedimientos tiene incorporado nociones de secuencia, efectos laterales y de términos normalmente necesarios en las tareas reales y de secuenciales.

La Fig. 4.2 [4] muestra una clasificación de la representación de conceptos en este marco de trabajo.

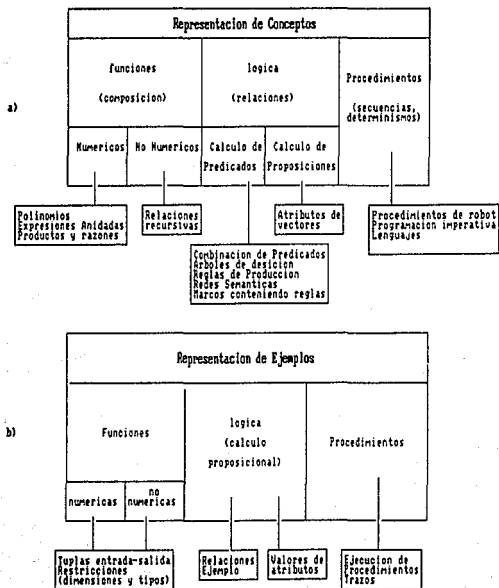


Fig 4.2 a) Marco de trabajo para la representación de conceptos.

b) Marco de trabajo para la introducción de ejemplos.

## LOGICA

Existen muchos métodos de aprendizaje que utilizan ejemplos que pueden ser expresados como vectores de atributos en forma equivalente al cálculo proposicional.

El cálculo de primer orden permite la existencia de términos variables en oraciones lógicas, y cuantificación sobre sus variables. Los vectores de atributos no cuentan con lo suficiente para describir situaciones en las cuales un ejemplo consiste de una escena con muchos ejemplos. Los objetos son caracterizados por sus atributos y pueden existir relaciones entre dos o más de ellos, esto implica que las variables deben ser introducidas para soportar objetos en varios tipos de relaciones. Estas relaciones pueden ser descritas como predicados que pueden ser nominales, lineales o con estructura de árbol al igual que los atributos. Los objetos y conceptos son caracterizados por combinaciones de predicados.

## FUNCIONES

Las expresiones funcionales incluyen muchas leyes naturales así como relaciones entre cantidades sensadas por un robot y parámetros de movimientos subsecuentes.

Las representaciones funcionales son apropiadas para expresiones anidadas y expresiones recursivas numéricas y no numéricas, y aunque es claro que cualquier representación funcional  $f(x)$  puede ser representada en lógica como para todo  $x$  existe  $y$ ,  $y = f(x)$  y son expresivamente equivalentes, en un marco de trabajo para inducción, es preferible tratar las expresiones funcionales de manera separada y omitir los cuantificadores explícitos.

## PROCEDIMIENTOS

Los conceptos típicos en esta categoría incluyen procedimientos de robots para ensamble, soldadura, etc. y procedimientos de oficina estándares como ordenar la correspondencia.

El formalismo de los procedimientos, es adecuado para la representación de ejecuciones secuenciales donde existen efectos laterales como asignación de variables y salidas reales como los movimientos de un robot, lo que hace que sea indispensable que el procedimiento se ejecute en el orden correcto.

Para describir formalmente un lenguaje procedural, se necesita un modelo del medio ambiente de ejecución en lugar de un modelo de substitución simple el cual sería suficiente para representaciones puramente funcionales. Las representaciones deben ser determinísticas para poder ser conceptos procedurales útiles. Cada decisión - orden-dependencia, determinismo - tiene efectos masivos en el espacio de búsqueda cubierto por una búsqueda inductiva.

La generalización de la ejecución de trazos en un procedimiento no es reducible a un problema equivalente que involucre la generalización de pares entrada-salida pues una reducción pierde información sobre los cambios secuenciales en un estado dado.

Existe un traslape obvio entre la representación lógica y la funcional no numérica. La diferencia consiste en que la lógica expresa la relación pura sin importarle las entradas y salidas, y la funcional actúa sobre una lista de entrada y produce una lista de salida. Por ejemplo el concepto de adicionar un elemento a una lista puede ser representado en cualquiera de los dos estilos.

#### B.- Representación de ejemplos.

La representación de ejemplos está estrechamente relacionada a la representación de conceptos para ejemplos que dirigen la búsqueda. En la Fig. 4.2 se puede observar que sus clasificaciones son de manera similar.

La diferencia entre ambos radica en que:

- Los ejemplos son generales. Se permite variables en los conceptos pero no en los ejemplos. De esta manera para las funciones, los ejemplos son tuplas entrada-salida de constantes. Para las relaciones, son tuplas que consisten de atributos de valores o relaciones sobre constantes y para los procedimientos son trazos de ejecución sin variables identificadas

- Si existen elementos opcionales en los objetos, solo una opción se cumple en cada ejemplo.

- Las estructuras de control procedural no ocurren en los ejemplos, de manera que los trazos de ejecución de ejemplos, son simples secuencias sin ramificaciones ni ciclos.

Los ejemplos de conceptos que son posibles de expresar en el cálculo de predicados, son expresables en el cálculo proposicional siempre y cuando no contengan variables.

### III. Tendencias.

La búsqueda es indispensable en todo aprendizaje de conceptos. La búsqueda involucrada en el aprendizaje inductivo así como en el deductivo no puede ser trabajada mientras no tenga fuertes tendencias de búsqueda ya que el espacio de posibilidades es muy grande.

Es muy importante en el aprendizaje de conceptos reconocer las tendencias que incorpora. La Fig. 4.1 muestra dos categorías principales

#### A. Conocimiento Base o Antecedentes.

Es importante mencionar que el sistema necesita tener ciertos conocimientos antes de poder inferir conceptos a partir de ejemplos, es decir, para aprender algo es necesario conocerlo primero. La cantidad de conocimiento que ya se posee sobre el dominio de un problema afecta de manera crítica el tipo de cosas que se espera sean aprendidas.

Las técnicas de aprendizaje de conceptos se pueden clasificar en las de conocimiento esparcido y conocimiento rico, no tanto por estar basados en la cantidad de información que incorporan, sino en la extensión en la cual el conocimiento es representado explícitamente.

Dado que el conocimiento base puede ser representado de muchas maneras, es difícil categorizar sus roles de la búsqueda con alguna tendencia.

La clasificación que se da a continuación está basada en la cantidad de información representada explícitamente y está hecha de manera que sea particularmente relevante para aplicaciones de Adquisición del Conocimiento.

*a) Técnicas de conocimiento esparcido.*

Este método es utilizado cuando un pequeño o ningún dominio de conocimiento está explícitamente disponible durante el aprendizaje. Existen tres categorías:

1) Aprendizaje de parámetros. Optimiza parámetros numéricos de un modelo pre-especificado. Estos sistemas son generalmente hechos para detectar regularidades en situaciones conflictivas y utilizan el conocimiento en la forma de técnicas estadísticas. La estructura del modelo no está representada explícitamente y el sistema no puede modificarlo o razonar sobre él. El aprendizaje de parámetros es un caso marginal del aprendizaje de conceptos con una base bien desarrollada de estadísticas.

2) Aprendizaje basado en similitudes.- Delinea el espacio de posibles descripciones de conceptos y los que mejor caracterizan las similitudes estructurales y/o las diferencias entre los ejemplos presentados.

3) Aprendizaje Jerárquico. Incrementa el lenguaje de descripciones a medida que cada concepto es aprendido. Permite que nuevas descripciones sean construidas sobre las viejas, de tal manera que el conocimiento base crece. La búsqueda está controlada por la selección apropiada del orden en el cual los conceptos son aprendidos. Es por esta razón que el maestro juega un papel importante en direccionar el aprendizaje.

*b) Técnicas de conocimiento rico.*

Estas técnicas utilizan mucho conocimiento sobre el dominio que se está trabajando. Trabajan relacionando la información nueva con el conocimiento base. Existen dos categorías:

1) Aprendizaje basado en explicaciones. Asimila una gran cantidad de información a través de un análisis intensivo de un solo ejemplo. Las técnicas deductivas son utilizadas para operacionalizar la parte de la base de conocimientos que se relaciona con los ejemplos. De esta manera el sistema no aprende conocimiento nuevo, solo articula reglas que ya están implicadas en la base de conocimientos.

2) Aprendizaje por descubrimiento. Es cuando el sistema opera autónomo realizando experimentos para mejorar y aumentar su base de conocimientos. Puede proceder de manera inductiva o deductiva. Este busca cosas interesantes, formando conjeturas al examinar algunos ejemplos de conceptos y notando coincidencias, o tratando de probar estas conjeturas.

### B) Inducción Obligatoria.

El aprendizaje de conceptos es más fácilmente manejable forzando la búsqueda a excluir grandes porciones del espacio potencial.

A continuación se presentan tres maneras diferentes en las cuales se puede forzar la inducción:

#### a) *Influencia conceptual.*

Limita el vocabulario para expresar los conceptos. Se ajusta muy bien tanto para las representaciones del cálculo de predicados, como funcional y procedural.

Este vocabulario no es el conjunto completo de símbolos de los cuales se construyen los conceptos, sino solo aquellos que aparecen en los ejemplos.

En la representación funcional la influencia conceptual determina cuales son las entradas (variables) y cuales son las salidas (resultados) de las expresiones.

#### b) *Influencia en la composición.*

Determina de que manera el vocabulario restringido puede ser utilizado para construir conceptos. Limita los conectivos usados cuando se forman expresiones a partir del vocabulario.

En las representaciones funcionales, la influencia en la composición determina el vocabulario de las funciones construidas, y los operadores y la sintaxis por medio de las cuales los conceptos podrán ser construidos.



La influencia en la composición en las representaciones procedurales es reflejada en el uso de un limitado repertorio de estructuras de control.

*c) Ordenación por preferencia.*

Cualquier observación puede ser explicada por un número indefinido de hipótesis. Las hipótesis no pueden ser justificadas intuitivamente, por lo que son inducidas de ejemplos observados. Los métodos para determinar el orden de preferencia dependen inevitablemente de sintaxis del lenguaje utilizado para expresar conceptos.

#### IV. Interacción con el Maestro.

La enseñanza difiere de la programación en que el profesor no utiliza un modelo formal del aprendiz, mientras que un programador normalmente espera saber exactamente como sus instrucciones serán interpretadas. Para ser un buen maestro no se necesita saber nada acerca de la estructura interna del aprendiz ni acerca de la representación que utiliza para los conceptos.

La gente halla difícil transferir su expertez en descripciones explícitas. Por lo que los sistemas de aprendizaje de conceptos deben estar capacitados para trabajar con el tipo de ejemplos que el maestro provee de manera natural, ordenados en una forma que es natural para él.

Existen tres aspectos importantes de la interacción del maestro con el sistema de adquisición de conocimientos:

##### A) Presentación.

Los ejemplos deben ser presentados incrementalmente o por lotes. Puede ser que la descripción actual deba ser incluida a medida que cada ejemplo sea encontrado, o alternativamente procesar todos los ejemplos juntos.

##### B) Ejemplos.

¿ Qué ejemplos son provistos ?  
Los ejemplos son la mayor y única información específica que el sistema tiene para ayudar a identificar un concepto en particular, por lo que la elección es crucial.

Una pregunta importante es: ¿El sistema puede generar sus propios ejemplos, los importantes para su propia estrategia de búsqueda de conceptos?. El aprendiz tiene la oportunidad de diseñar casos de prueba para hipótesis específicas sobre el concepto que se está tratando, mientras que los ejemplos provistos por el profesor pueden ser irrelevantes.

Una segunda pregunta es : ¿Que tan confiables son los ejemplos? Estas pueden contener componentes "ruidosos" o ser mal clasificados. Muchos sistemas de aprendizaje de conceptos asumen que los datos son exactos de manera que las regularidades estructurales pueden ser aprendidas sin problemas de detección estadística.

#### C) Qué tan bueno es el profesor.

Esta tercera área concierne a la cooperación y habilidades pedagógicas del profesor.

El rango de situaciones de aprendizaje desde no tener un profesor (las entradas son provistas directamente del medio ambiente) a través de un simple usuario, un experto del dominio, un profesor entrenado y hábil, hasta tener un profesor que está preparado para programar en un lenguaje de programación convencional y con ello evitar la necesidad de aprendizaje.

Un profesor hábil seleccionará ejemplos demostrativos para el y con ello simplificará la tarea del aprendiz. El profesor debe clasificar los ejemplos correctamente, debe ser explícito, evitar rodeos sobre resultados inmediatos e introducir esencialmente una característica por lección. Los ejemplos no deben ser coincidencias que incluyan características que puedan confundir al aprendiz.

La posibilidad de que un sistema construya sus propios ejemplos y los tenga clasificados por un informante, tiene un potencial considerable para agilizar el aprendizaje y reducir la dependencia en la habilidad del profesor.

### V. Uso de este marco de trabajo.

Dado un concepto deseado y representaciones de ejemplos, conocimiento base disponible, forma de interacción con el maestro e influencias apropiadas para la representación de conceptos, no es posible reemplazar un conjunto de reglas comprensivas para determinar que esquema de aprendizaje de conceptos se debe utilizar.

A continuación se indica por medio de ejemplos de tipos particulares de sistemas, cómo elegir adecuadamente un esquema de Adquisición de Conocimientos. Dado un problema práctico de adquisición de conocimientos, la primera decisión a tomar es como representar los conceptos y ejemplos. Algunas formas adecuadas de representación de conceptos están dictadas por los requerimientos del Sistema Experto y el tipo de ejemplos disponibles. En algunas ocasiones la representación de los ejemplos predomina en la decisión, mientras que en otras situaciones el formato deseado para los conceptos fuerza a los ejemplos a tener un molde o forma particular.

A continuación se describen algunas representaciones de conceptos basadas en la lógica diferenciadas por la manera en la que el conocimiento base puede influenciar la búsqueda: basada en similitudes, jerárquica, basada en explicaciones y basada en descubrimientos. Posteriormente se describen otras representaciones basadas en funciones y procedimientos.

Las representaciones lógicas están determinadas por el predominio de relaciones lógicas en los ejemplos o en los conceptos, por ejemplo la posibilidad de utilizar valores de los atributos sugiere el uso de una representación en el cálculo proposicional.

Los métodos de sistemas basados en similitudes son apropiados cuando se tienen disponibles muchos ejemplos o cuando no es posible definir a priori una teoría del dominio. Por otra parte, si los conceptos deben ser construidos sobre los aprendidos con anterioridad, el método jerárquico es el apropiado. Pero si la teoría del dominio es conocida, entonces se puede utilizar un sistema basado en descubrimientos o uno basado en explicaciones.

**A.- Representación lógica: Basada en similitudes.**

Dado un conjunto de objetos que representan ejemplos y contraejemplos de un concepto, el aprendizaje basado en similitudes trata de inducir una descripción generalizada que abarque a todos los ejemplos y a ningún contraejemplo. El interés en los asuntos generales incluye las condiciones bajo las cuales el procedimiento converge a una descripción simple, donde el sistema puede saber si ha convergido, donde el concepto final puede depender del orden de presentación de los ejemplos y donde se espera que el conjunto de formación sea exhaustivo o representativo. A continuación se presentan algunos ejemplos.

**1.- Espacio de Versiones ( Version Space).**

La aproximación del espacio de versiones al aprendizaje de conceptos, transforma el problema inductivo de la generalización en uno deductivo, esto se logra circunscribiendo la manera en la cual las descripciones son expresadas, así como la manera en la que se busca que algunas encajen con los ejemplos dados. Esta aproximación postula un lenguaje en el cual los objetos son expresados, un lenguaje en el que los conceptos son expresados, un predicado que determina si un objeto particular encaja con una descripción particular, y un ordenamiento parcial sobre las descripciones (interpretado como generalización/especialización). Dado un conjunto de ejemplos positivos y negativos de una descripción particular, existe un algoritmo simple de búsqueda que encuentra todas las descripciones que son consistentes con los ejemplos. Este conjunto es llamado espacio de versiones.

Este método es aplicado de manera simple y directa cuando cada objeto es descrito por un conjunto fijo de propiedades, usualmente representado como un vector de atributos el cual es equivalente a una descripción en la lógica proposicional.

En dominios estructurados cada ejemplo (o contraejemplo) comprende una "escena" que contiene muchos objetos en la lógica de predicados. Parte del problema en la igualación de la escena con una descripción estructural está en determinar un mapeo apropiado entre los objetos en la escena y aquellos especificados en la descripción. Este

mapeo tendrá diferentes interpretaciones dependiendo si la escena es el objeto mismo o simplemente contiene al objeto deseado. Existe muchos resultados teóricos que indican que incluso casos simples involucran complejidades computacionales extremas al trabajar con espacios de versiones de objetos estructurados.

## 2.- Creación de Reglas de decisión.

Se puede construir un árbol de decisión o una colección de reglas que discrimine los ejemplos positivos de los ejemplos negativos. En contraste con el método de espacio de versiones, este asume que todos los ejemplos están disponibles y que pueden ser procesados juntos.

En el caso de un árbol de decisión, la raíz especifica un atributo a ser seleccionado y probado, luego dependiendo de su valor los nodos subordinados determinan la prueba para los subsecuentes atributos. Las hojas son marcadas para mostrar las clasificaciones de los objetos que representan. Para problemas biclasificados, estas clasificaciones son simplemente: "positivo" y "negativo", pero es fácil distinguir si se trata de mas de dos clases. El algoritmo ID3 utiliza una heurística información-teoría para encontrar un solo árbol que clasifique todos los ejemplos dados. Este procedimiento ha sido estudiado extensivamente y adoptado para algunas aplicaciones comerciales. Cuando se presentan datos "ruidosos", este procedimiento construye grandes árboles de decisión que reflejan el detalle de todos los ejemplos vistos. <como es de esperarse, la representación resultante no es económica y no es un medio adecuado para la generalización.

En el caso de las reglas de producción, el conjunto de formación es utilizado para construir un conjunto de reglas que pueda ser interpretadas por un Sistema Experto en una manera de encadenamiento estandard hacia adelante o hacia atrás. Aunque cualquier árbol de decisión puede ser fácilmente convertido a reglas, estas pueden contener redundancias que pueden ser eliminadas si se generan directamente de los ejemplos.

## B. Representación lógica: Aprendizaje jerárquico.

Sería atractivo si los sistemas de aprendizaje pudiesen construir conceptos sobre aquellos aprendidos con anterioridad y utilizarlos como componentes en las

descripciones mas recientemente construídas. Esto posiblemente permitiría que el aprendizaje sea sostenido sobre un periodo extendido de tiempo, en lugar de ser realizado en base a un solo tiempo.

### 1.- Arboles de Conceptos.

Existe un sistema llamado Marvin que aprende estructuras jerárquicas de conceptos. Dado un ejemplo, este busca los caminos para expresarlo en términos de los conceptos conocidos. En lugar de esperar ejemplos nuevos del profesor para forzar la elección de las expresiones, éste selecciona una descripción tentativa, sintetiza un objeto crucial para determinar si esta descripción es una generalización correcta y prueba si esta es también un ejemplo de un concepto. Si es, la descripción tentativa es aceptada y además se intenta generalizarla. Si no es, la descripción tentativa es especializada para excluir un ejemplo negativo y un nuevo ejemplo crucial es sintetizado.

De esta manera, se realiza una secuencia de especializaciones y generalizaciones. Cada una es probada pidiendo al profesor que clasifique un ejemplo crucial. Una vez que se han tomado todas las posibilidades de generalización, la descripción es almacenada y se le pide al profesor otro ejemplo de un concepto.

### 2.- Redes Semánticas.

El método implementado por Marvin no funciona si los conceptos no están estructurados y enseñados en forma de árbol. Un caso extremo de una representación no jerárquica es cuando cada objeto es representado por un vector de atributos. Las bases de conocimiento del mundo real, por lo general caen entre los extremos de una representación jerárquica y una representación competentemente plana. Marvin ha sido extendido para tratar apropiadamente tales situaciones, este realiza investigaciones minuciosas de hipótesis competitivas en cada estado. Si un objeto crucial existe dentro de la hipótesis tentativa actual pero no dentro de las competitivas, es presentada al profesor. Si tal objeto no existe, las hipótesis alternativas son investigadas para descubrir la descripción válida mas general.

### C. Representación Funcional.

La representación de conceptos como funciones, es apropiada cuando los ejemplos toman la forma de pares entrada-salida y el conocimiento a ser adquirido es una expresión para calcular la salida de cada par a partir de su entrada. Para las funciones numéricas, se puede utilizar aproximaciones polinomiales. Sin embargo esto no es tan útil como parece a primera vista. Por una parte, el problema de la complejidad surge nuevamente. Mientras que para algunos puntos de vista una función trascendental como  $\sin x$  puede ser adecuadamente aproximada por un polinomio de orden mayor, usualmente existen ventajas si en su lugar se utiliza una descripción más concisa.

1. Marvin. Puede ser utilizado para inducir algunas relaciones lógicas debido a las funciones. Los ejemplos de conceptos que ha aprendido incluyen la manipulación de listas, aritmética, y un ordenamiento de inserción simple.

2. Bacon. Es un sistema de Inteligencia Artificial que trabaja con todas las funciones numéricas. Trata de descubrir leyes científicas empíricas induciendo funciones que contabilizan los datos observados. Como entrada se le proporcionan una tabla de valores de las variables dependientes y de las independientes y busca una expresión funcional usando subexpresiones teóricas para direccionar la búsqueda.

3. Coper. Una aproximación más estructurada para deducir leyes físicas, es utilizar análisis dimensional en conjunto con los datos observados. Coper trabaja con cantidades físicas y utiliza análisis dimensional para determinar si las variables independientes son suficientes para determinar los valores únicos de las variables dependientes. Selecciona una base dimensional apropiada y la utiliza para generar expresiones polinomiales.

### D. Representaciones Procedurales.

Los conceptos son mejor representados como procedimientos, si los ejemplos son trazos secuenciales de las acciones requeridas para realizar alguna tarea. Al igual que otros problemas inductivos, la adquisición de procedimientos a partir de ejemplos es en general intratable debido a los enormes espacios involucrados. Los sistemas prácticos deben

implementar métodos para limitar la búsqueda basada en el conocimiento del dominio.

Noddy adquiere los procedimientos de un robot y los complementa con información de control que no está explícitamente presente en los ejemplos. Este cubre problemas de secuencialización de acciones y también maneja los números reales que presentan ángulos de distancias. Este emplea una generalización jerárquica, explícita y preprogramada e información preprogramada sobre 30 operadores. En Noddy los ejemplos son trazos del ejemplo deseado. El primer trazo es tomado como la versión inicial del procedimiento. Dado que los trazos nuevos son vistos a medida que son unidos al procedimiento que está naciendo, este se generaliza de varias maneras.

Este sistema no puede reconsiderar generalizaciones que ha realizado en la versión actual del procedimiento, y por lo tanto adopta una política conservadora que requiere un cantidad considerable de evidencias antes de realizar la generalización.



## II. ADQUISICIÓN DEL CONOCIMIENTO PROCEDURAL DE LOS EXPERTOS DEL DOMINIO.

El conocimiento del experto se compone de dos partes: el conocimiento declarativo y el procedural. Las técnicas anteriormente mencionadas tratan de la adquisición del conocimiento declarativo. Sin embargo una parte importante de la expertez la forma el conocimiento procedural que se conforma de muchos elementos, desde reglas de bajo nivel para la manipulación de datos, hasta estrategias abstractas para la resolución de problemas.

### **Tipos de conocimiento procedural.**

Las clases de conocimiento procedural dependen básicamente del dominio de expertez alrededor del cual se construirá el Sistema Experto, algunas de las clases principales son [5]:

- Procedimientos para la manipulación de datos.
- Procedimientos para la simulación.
- Estrategias de diseño de experimentos.
- Heurística para la selección.

Los Procedimientos para la manipulación de datos analizan la información existente en la Base de Conocimientos para extenderla o corregirla si fuera necesario. La mayor parte consta de reglas que aseguran la consistencia y completéz de la Base de Conocimientos. Los expertos del dominio describen estos procedimientos para inferir alguna información y son utilizados siempre que se introducen nuevos elementos en la Base.

Los Procedimientos de simulación modelan las acciones de las diferentes operaciones realizadas en el dominio. Contiene la información necesaria para alterar la representación de objetos cuando un operador es hipotéticamente aplicado.

La heurística para la selección es utilizada para la selección de una operación entre varias operaciones alternativas competitivas durante el proceso de diseño de experimentos.

**Las estrategias de diseño de experimentos son el conocimiento sobre métodos de resolución de problemas. Este conocimiento puede estar a muchos niveles de abstracción e independencia del dominio.**

**Adquisición del conocimiento directamente de los expertos del dominio.**

Dados estos tipos de conocimiento procedural, el problema al que nos enfrentamos es cómo adquirirlos. La manera típica, como ya se mencionó, es a través de entrevistas entre el Ing. del Conocimiento y el o los expertos del dominio. Por otra parte, como se realizó en el sistema MOLGEN [5], se enfatiza en que los mismos expertos del dominio construyan la Base de Conocimientos. Bajo este método de trabajo, la labor del Ing. del Conocimiento es proporcionar las herramientas de representación apropiadas tanto para el conocimiento declarativo como procedural del dominio.

Este método se considera de gran importancia pues la precisión y completez de la Base de Conocimientos es arriesgada cuando el conocimiento del dominio pasa a través de un filtro: el Ing. del Conocimiento, quien no es un experto del dominio en el que se está trabajando. La mayor parte del conocimiento es complejo y sutil y puede que su propósito no sea significativo para una persona no experta.

La construcción de gran parte de la Base de Conocimientos puede verse retardada si personas no-expertas tiene que describir cada objeto y regla que la conformará. Es obvio que existen ciertas desventajas y prerequisites si se permite a los expertos describir su conocimiento en una Base de Conocimientos. Muchos Ing. del Conocimiento que han colaborado con los expertos del dominio en esta labor aseguran que existen pocos expertos que inicialmente tienen la habilidad de formalizar lógicamente las reglas de su dominio [5]. Por lo antes mencionado, los Ing. del Conocimiento son de gran utilidad en la organización lógica de las reglas del experto. Por lo general los expertos se vuelven más formales a medida que aumenta su experiencia con los Sistemas Expertos.

**Método para la descripción del Lenguaje Procedural: "EL LENGUAJE DE REGLAS".**

Una manera mediante la cual los expertos del dominio podrían introducir conocimiento procedural es aprendiendo el lenguaje de programación que utiliza el Sistema Experto, pero tiene sus desventajas pues por una parte se necesita invertir mucho tiempo para enseñar al experto a ser un buen programador, y por otra parte se violaría el principio de mantener el sistema transparente para el usuario.

Una solución posible para permitir la descripción de todos los tipos de conocimiento procedural es proveer un lenguaje "procesado" del "inglés genérico" como un lenguaje de programación simplificado. Este lenguaje permitirá a los expertos del dominio acceder y manipular varias unidades dentro de sus bases de conocimientos es decir "variables de su lenguaje de programación". Gran parte de este lenguaje es independiente del dominio, como se verá a continuación, sin embargo su arquitectura permite que existan partes que estén especialmente diseñadas de acuerdo al dominio de la expertez.

Como se menciona en [5], las raíces de esta idea, de contar con un lenguaje especializado para obtención de la expertez procedural, nace con el proyecto MYCIN donde se utilizó un lenguaje formalizado para la escritura de las reglas de producción con la finalidad de introducir al sistema conocimiento experto sobre diagnósticos de enfermedades infecciosas. De esta manera estas ideas fueron madurando hasta convertirse en un método formal.

**Diseño General.**

Este lenguaje para la descripción del conocimiento procedural puede ser implementado como un Lenguaje de Reglas. Los mecanismos de interacción con este son básicamente los mismos que los que se utilizan con el conocimiento declarativo. El usuario interactúa con un editor que le proporciona herramientas para la introducción y corrección de oraciones individuales dentro del lenguaje.

Las reglas introducidas son revisadas e interpretadas en dos pasos. Al momento de ser introducidas, se revisa la sintaxis después de lo cual son almacenadas en un formato interno de procedimientos y parámetros. Al momento de ser interpretadas, se encuentran los parámetros obligatorios dentro del contexto y se toma la acción del procedimiento. Esto mismo permite que se escriban reglas sobre entidades que no existen aún en la Base de Conocimientos y permite que estas reglas sean generales para más de un contexto.

Estas reglas pueden ser de una de las siguientes formas

- Una acción. Algo que crea, borra o modifica una unidad, o que afecta la interpretación de un conjunto de reglas. Las acciones pueden ser unidas por la palabra "AND".
- Una oración condicional de la forma IF.. THEN .. ELSE donde la parte ELSE es opcional.
- Una oración iterativa de la forma "FOR descripción de la iteración - expresión booleana" o "FOR descripción de la iteración - acción".
- Una etiqueta, es decir un punto referencia indicado por la palabra "LABEL" seguido por una sola palabra.

#### Acciones.

Las acciones pueden estar clasificadas como independientes del dominio o específicas del dominio. Por ejemplo la acción independiente ADD adiciona un elemento a una lista.

#### Expresiones Booleanas.

Una expresión Booleana consiste de una o más relaciones que pueden ser combinadas con los operadores "OR", "AND" o "NOT"

#### Instrucciones Iterativas.

Esta permiten al usuario iterar sobre grupos de unidades, renglones de una tabla o elementos de una lista. El contexto es automáticamente actualizado durante cada paso de la iteración. Durante las

iteraciones sobre una tabla, los datos de entrada individuales pueden ser recuperados o alterados haciendo referencia a los nombres de las columnas.

#### Facilidades para hacer mas sencilla la escritura de la reglas.

A medida que el lenguaje de reglas a mejorado a través de los últimos años, también han surgido facilidades para programar en estos. Los usuarios pueden libremente hacer uso de sinónimos comunes y obviar palabras para hacer que las reglas sean mas legibles. Las reglas pueden ser introducidas directamente por medio de un editor de reglas que permita la inserción, el borrado y reemplazo de reglas.

Como se mencionó anteriormente la variables obligatorias no se establecen hasta el momento de la interpretación de las reglas de manera que la información de las variables sobre las que opera un conjunto de reglas no tiene que existir cuando las reglas son escritas.

La mayoría de los usuarios principales de los lenguajes de reglas, prefieren aprender a usar el sistema interactuando directamente con el, en lugar de tener que leer primero las instrucciones. Estos hacen uso de las facilidades y ayudas para el lenguaje de reglas que esta almacenado dentro de la base de conocimientos.

Realmente el lenguaje de reglas comienza solamente con el verbo SET, la estructura IF .. THEN .. ELSE, y las relaciones booleanas =, <, >.

Todas la estructuras relacionadas son creadas por los requerimientos de los usuarios. Por esto se puede decir que prácticamente son los usuarios quienes crean su propio lenguaje de programación.

## CAPITULO 5

### APLICACION

En este capítulo se presenta una descripción detallada de la aplicación de la tesis que consiste en la Adquisición del Conocimiento para un Sistema Experto de componentes reutilizables de Tipos de Datos Abstractos (TDA's).

Como se mencionó en la introducción, la técnica de Adquisición del Conocimiento que se utilizó durante la interacción con los expertos fue la de entrevista y durante la transcripción de las entrevistas, fue la técnica de Malla de Repertorio.

A lo largo de este capítulo se describe como se realizaron las entrevistas con los expertos, como se realizaron las interpretaciones de tales entrevistas, como se utilizó la técnica de Malla de Repertorio durante la transcripción de las entrevistas y como se llegó al planteamiento final.

#### I. Descripción del Sistema Experto.

El Sistema Experto deberá permitir al usuario almacenar las especificaciones, conceptos, aplicaciones, así como implementaciones de los componentes reutilizables de los TDA's en diferentes lenguajes de programación y deberá guiar al usuario en la selección de los componentes adecuados.

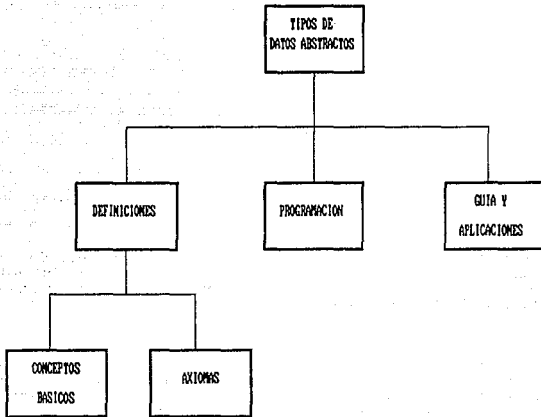


Fig. 5.1 Esquema de un Sistema Experto de componentes reutilizables de Tipos de Datos Abstractos.

Este sistema constará de varios módulos como se muestra en la Fig 5.1. Existirá un módulo de Conceptos en el que se proporcionará al usuario una descripción completa del TDA elegido. Un módulo de Axiomas en el que se presentará al usuario los axiomas que conforman el TDA. Un módulo de Programación, que para cada TDA, estará conformado de varios algoritmos para diferentes operaciones sobre los TDA's como inserción, eliminación, búsquedas, ordenamientos, etc., codificados en diferentes lenguajes de programación. Y finalmente, el módulo que es objeto de nuestro interés y aplicación que es el de Guía y Aplicaciones, en el que se plantea al usuario una serie de preguntas por medio de las cuales se le guiará hacia el componente adecuado.

Estos módulos deberán estar interconectados de tal manera que una vez que el usuario este posicionado sobre un componente dado, pueda revisar sus conceptos, axiomas y algoritmos.

## II. Adquisición del Conocimiento.

La Adquisición del Conocimiento para la presente aplicación se enfoca a obtener de los expertos en TDA's, las preguntas que permitirán guiar a un usuario no experto hacia los componentes adecuados.

Los expertos que me ofrecieron su colaboración para la realización de esta aplicación son el Dr. Mario Magidín, el Ing. Luis Fernando Castro y la Dra. Hanna Oktaba Coordinadora de la Maestría en Ciencias de la Computación del I.I.M.A.S quien es autora de todas las ideas que conforman la realización del Sistema Experto.

A continuación se presentan las entrevistas que realicé con los expertos. Lastimosamente estas entrevistas fueron pocas y cortas debido a las múltiples actividades que desempeñan estas personas, a quienes agradezco enormemente el haberme dedicado un poco de su valioso tiempo.

Las entrevistas se describirán con un formato de pregunta y respuesta, donde la pregunta es textual como se realizó a los expertos y la respuesta es una descripción lo más aproximada posible a las respuestas del experto.

### **Primera y única entrevista con el Dr. Mario Magidín.**

En la entrevista que sostuve con el Dr. Magidín, inicialmente le expliqué en que consistiría el Sistema Experto, cuales eran sus objetivos y cual era el papel que yo deseaba que él desempeñara dentro de este.

A su vez el Dr. Magidín expresó su gran interés en el proyecto, pero aclaró que disponía de muy poco tiempo para dedicarle a esta actividad por lo que decidió empezar inmediatamente con la entrevista.



Pregunta: ¿Cómo guiaría usted a un usuario que no tenga mucho conocimiento sobre TDA's?

Respuesta: En respuesta a tal pregunta, él afirmó que un aspecto importante es determinar si se desea tener acceso aleatorio, y si la estructura es unidimensional o jerárquica.

Pregunta: ¿En caso de que el usuario afirmara que desea tener acceso aleatorio en el TDA, que le sugeriría usted?

Respuesta: A lo que respondió que si existiesen en cantidad inserciones o remociones aleatorias, él sugeriría listas ligadas y que si los accesos a los elementos del TDA fueran aleatorios y en gran cantidad, él sugeriría una implementación con arreglos.

Adicionalmente a lo anterior, afirmó que si los accesos fuesen por un solo extremo, sugeriría una pila, y en el caso de que las remociones se realicen por un extremo y las inserciones por el otro, se debía sugerir el uso de una cola, y que recomendaría una implementación con listas ligadas si los acceso solo se hacían al tope de la pila y con arreglos si los demás elementos de la pila también se revisaban con cierta frecuencia.

Con esto último, finalizó la entrevista con la esperanza de poder seguir con el tópico en una entrevista próxima después de haber revisado bibliografía referente al tema. Esto último no sucedió pues no pude volver a entrevistar al Dr. Magidín y únicamente obtuve de él una lista con la bibliografía que él me recomendaba revisar para obtener la información que necesitaba.

**Primera entrevista con el Ing. Luis Fernando Castro Careaga.**

En esta entrevista inicié el tópicó describiendo al Ing. Castro las características y objetivos del Sistema Experto así como la finalidad y metas de la entrevista, es decir la Adquisición del Conocimiento que el Ing. poseía para guiar a un usuario no experto en la elección de un TDA adecuado.

**Respuesta:** El Ing. Castro inmediatamente respondió que el propondría niveles de preguntas que guiarían de alguna manera ordenada hacia el componente adecuado.

**Pregunta:** ¿A que se refiere con niveles de preguntas y en que consistirían estas?

**Respuesta:** Su respuesta fue que con estos niveles se irían estableciendo el tipo de componentes a utilizar y que estos serían:

- 1.- Homogeneidad de la información.
- 2.- Tiempo de vida del TDA.
- 3.- Forma de acceso a sus elementos.

en ese orden.

**Pregunta:** ¿Con estos niveles establecería una especie de jerarquía con la que a medida que se avance de niveles, se irían eliminando posibles TDA's?

**Respuesta:** A lo que el respondió que precisamente ese era el objetivo de establecer niveles.

**Pregunta:** ¿Podría describir estos niveles?

Respuesta: La descripción que proporcionó fue la siguiente :

1.- El nivel de Homogeneidad de la información tiene por objetivo establecer el tipo de lenguaje a utilizar, pues no todos permiten la heterogeneidad de la información en un mismo TDA.

2.- En el nivel de Tiempo de vida, mencionó que era necesario establecer si el TDA duraría más allá de la duración de un programa, pues en tal caso la información sería almacenada en archivos y habría que recomendar un tipo de TDA apropiado para el manejo de archivos.

3.- En el nivel de formas de acceso, mencionó que en el caso de trabajar con archivos, era necesario establecer si los registros deberían estar clasificados.

Adicionalmente mencionó la importancia de la organización de la información al realizarse operaciones de inserción y eliminación sobre el TDA, puesto que si se establecía el lugar donde iban a quedar los elementos insertados o de donde se deberían tomar los eliminados, se podría establecer si tendrían un orden temporal o lógico con lo cual se podría definir si se recomendaría pilas, colas, o listas.

Pregunta: ¿Estos niveles de preguntas son suficientes para determinar el TDA adecuado?

Respuesta: A lo que él respondió que tal vez faltarían algunos factores como ubicar una pregunta que establezca si la información tendrá el mismo nivel o existirá una jerarquía.

Pregunta: ¿En caso de ser jerárquica, que tipo de TDA recomendaría?

**Respuesta:** Su respuesta fue que dependía de muchos factores como los que ya había mencionado y otros como la cantidad de niveles de la jerarquía y que se podría recomendar el uso de arboles y/o tablas.

**Pregunta:** Mencionó que si la estructura permaneciera mas allá de la vida de un programa, ud. recomendaría una estructura apropiada para el manejo de archivos, ¿A que se refería?

**Respuesta:** A lo que él respondió que recomendaría el uso de arboles AVL en caso de que la información deba reorganizarse con frecuencia. Recomendó el uso de arboles B en caso de requerir operaciones muy rápidas, en caso de tener mas de 100 registros y que la reorganización tenga características locales.

Con esto terminó la entrevista, debido a que sentimos la necesidad de revisar algo de bibliografía, de manera que se pudiera reorganizar la información que se había expuesto hasta el momento.

#### Transcripción de las dos entrevistas.

Después de estas dos entrevistas, realicé la transcripción de estas con la finalidad de establecer un orden en toda la información recibida, hacer un planteamiento de las preguntas que podrían guiar al usuario hacia los componentes adecuados y para establecer las dudas, preguntas y tópicos a tratar en la siguiente entrevista con el Ing. Castro.

Después de analizar la información obtenida, decidí proponer una estructura jerárquica de preguntas que a mi juicio guiarían adecuadamente a un usuario no experto.

En cada nodo de la estructura jerárquica que se muestra en la Fig. 5.2 existe una pregunta cuyas respuestas solo pueden ser 2 : "si" o "no", "temporal" o "lógico", etc., de manera que obtuve el siguiente árbol binario con sus correspondientes preguntas.

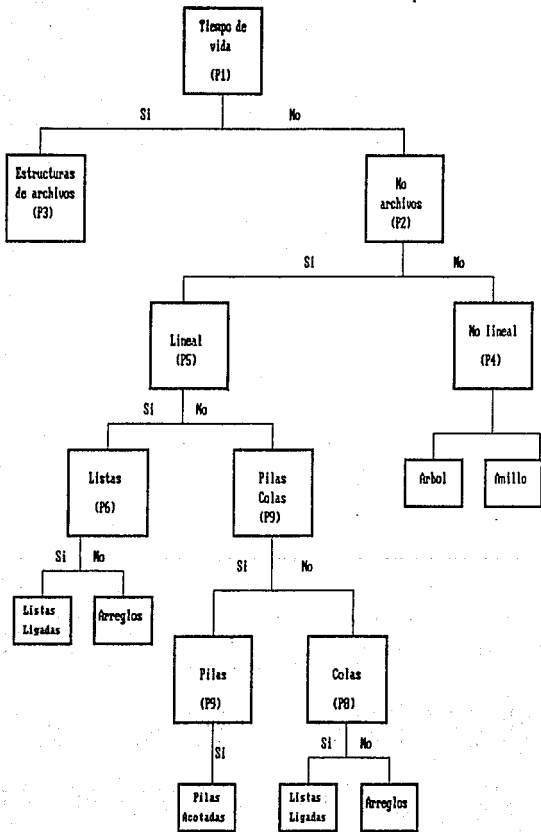


Fig. 5.2 Estructura binaria para la seleccion de TDA's.

**Preguntas**

- P1.- ¿El tiempo de vida de la estructura es mayor a la duración de un programa? o ¿La estructura debe persistir al finalizar el programa?.
- P2.- ¿La estructura debe ser lineal y poseer elementos límites definidos?
- P3.- ¿Los registros deben estar clasificados?.
- P4.- ¿Debe establecerse una jerarquía en la estructura?.
- P5.- ¿En la operación de inserción o eliminación de un elemento, este debe tener dentro de la estructura un orden lógico o temporal?
- P6.- ¿Las operaciones de eliminación e inserción son más frecuentes que las búsquedas?
- P7.- ¿El último elemento que se introduzca, será el primero en eliminarse? o ¿ las eliminaciones e inserciones deben realizarse por un solo extremo?
- P8.- ¿Se revisan con frecuencia, a modo de consulta, los elementos de la cola?
- P9.- ¿Está limitada la cantidad de elementos que se insertarán en la pila? o ¿está limitada la cantidad de memoria que se utilizará para almacenar la pila?

**Segunda entrevista con el Ing. Luis Fernando Castro.**

En la segunda entrevista con el Ing. Castro, le planteé el esquema que había obtenido durante la transcripción. La respuesta que obtuve de él fue que efectivamente debería ser una estructura jerárquica la que nos conduciría hacia los componentes correctos, pero que un árbol binario no nos conduciría a la respuesta correcta, pues la mayor parte de las veces iba a ser un conjunto de características las que nos acercarian al componente adecuado y no una sola.

Para aclarar lo anteriormente mencionado, me planteé dos ejemplos con las siguientes preguntas, respuestas, y solución final.

**Ejemplo 1.-**

Pregunta: ¿Se desea una estructura jerárquica?

Respuesta: Si.

Pregunta: ¿De cuantos niveles es la jerarquía?

Respuesta: de 2.

Pregunta: ¿Sus inserciones son respecto al tiempo o deben preservar un orden lógico?

Respuesta: Deben preservar un orden lógico.

Recomendación Final: Dos tablas con llaves.

**Ejemplo 2.-**

Si el usuario busca velocidad sin importar otras características, él recomendaría el uso de árboles B y si necesita una mayor velocidad aún, él recomendaría el uso de tablas Hash.

De esta manera finalizó la segunda y última entrevista con el Ing. Castro a quien no pude volver a entrevistar por haberse ido de la ciudad.

Con toda esta información nueva decidí utilizar la técnica de malla de Repertorio con la finalidad de organizar las transcripciones e información obtenida.

### III. Uso de la técnica de Malla de Repertorio.

Como primer intento decidí proponer como constructores, 6 características que a mi parecer, sus valores deberían determinar los diferentes TDA's. Estas características las obtuve al analizar el árbol que había expuesto al Ing. Castro, y son las siguientes:

<u>Constructores</u>	<u>Posibles valores</u>
C1: Tiempo de vida	Duración de un programa, o mas de un programa.
C2: Estructura jerárquica	Si, No.
C3: Estructura Lineal	Si, No.
C4: Orden en la operaciones (inserción, eliminación)	Temporal, Ordenado lógicamente.
C5: Gran cantidad de accesos aleatorios	Si, No.
C6: Estructura Acotada	Si, No.

Con la combinación de todos estos posibles valores de estos constructores, construí una malla en la que cada combinación hacia referencia a un Elemento, que para este caso sería un TDA específico.



C1	Tiempo de Vida	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe	Pr	Pe		
C2	Estructura Jerárquica	SI	SI	No	No	SI	SI	No	No	SI	SI	No	No	SI	SI	No	No	SI	SI	No	No	SI	SI	No	No	SI	SI		
C3	Estructura Lineal	SI	SI	SI	SI	No	No	No	No	SI	SI	SI	SI	No	No	No	No	SI	SI	SI	SI	No	No	No	No	SI	SI		
C4	Orden en la operación	Y	Y	Y	Y	Y	Y	Y	Y	O	O	O	O	O	O	O	O	Y	Y	Y	Y	Y	Y	Y	O	O	O		
C5	Accesos aleatorios	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	No	No	No	No	No	No	No	No	No	No	No	No		
C6	Estructura Acotada	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI		
		E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14	E15	E16	E17	E18	E19	E20	E21	E22	E23	E24	E25	E26	E27	E28

Pr: Duración solo durante un programa.

Pe: Duración permanente.

Y: Orden temporal.

O: Orden lógico.

Fig 5.3 Malla de repertorio con 6 constructores.

Con todas las combinaciones, construí una malla que me proporcionaba 64 elementos. Una parte de tal malla se muestra en la Fig. 5.3. El problema que se presentó fue que existían combinaciones de características que generaban elementos que no hacían referencia a ningún TDA. Por ejemplo el elemento E1, me sugiere un TDA cuya vida termina al finalizar el programa, que debe ser jerárquico, y al mismo tiempo debe ser una estructura lineal (lo cual se contraponen), de manera que sin revisar los valores de los demás constructores, ya poseía el problema de no saber a que TDA podría corresponder el elemento E1.

Al enfrentarme a tal problema decidí revisar cuales eran las características que se contraponían y en que casos. Como resultado obtuve que:

1.- Cuando el constructor C1: Tiempo de Vida, tenía el valor de duración mayor a la de un programa, debería tener un tratamiento diferente pues las demás características no favorecían a la selección de un TDA.

2.- En el caso en el que el constructor C4: Orden en las operaciones, tenía el valor de ordenado lógicamente, no existían estructuras acotadas que correspondieran a tal característica.

3.- Y finalmente que en el caso de que el constructor C2: Estructura jerárquica tuviese el valor de Si, no existía ninguna estructura que estableciera un orden temporal durante las operaciones de inserción y eliminación de elementos.

Ante tales problemas decidí fraccionar la malla en dos partes según los valores de algunas características elegidas previamente. El resultado que obtuve se muestra a continuación en la Fig 5.4.

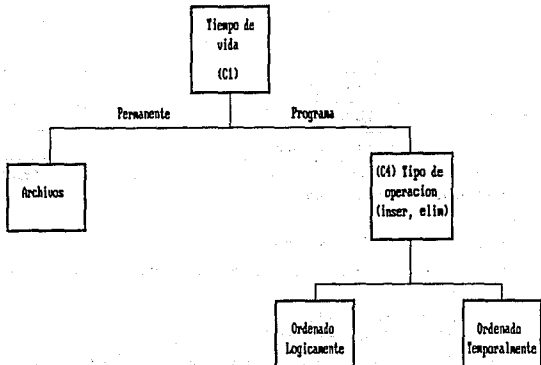


Fig. 5.4 (a).

		E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12	E13	E14	E15	E16
C1	Estructura lineal	S	N	S	N	S	N	S	N	S	N	S	N	S	N	S	N
C2	Acceso aleatorio	S	S	N	N	S	S	N	N	S	S	N	N	S	S	N	N
C3	Inserciones y elim. por un solo extremo	S	S	S	S	N	N	N	N	S	S	S	S	N	N	N	N
C4	Estructura Acotada	S	S	S	S	S	S	S	N	N	N	N	N	N	N	N	N
Pila acotada con arreglos																	
Pila acotada con listas ligadas																	
Cola acotada con arreglos																	
Anillo acotado con arreglos																	
Cola acotada con listas ligadas																	
Anillo acotado con listas ligadas																	
Pila con arreglos																	
Pila con listas ligadas																	
Cola con arreglos																	
Anillo con arreglos																	
Cola con listas ligadas																	
Anillo con listas ligadas																	

Fig 5.4 b). Malla adquirida para seleccionar TDA's con ordenamiento temporal.

**Ordenado Temporalmente.**

E5, E7, E13, E16: Colas

a) ¿Las eliminaciones e inserciones pueden ser por ambos extremos?

Respuesta : Si.  
Sugerencia: Bicolos.

b) ¿Los elementos deben ser almacenados juntos por prioridades?

Respuesta : Si.  
Sugerencia: Colas con prioridad.

Ordenado logicamente		E1	E2	E3	E4	E5	E6	E7	E8
C1	Estr. Jerarquica	S	N	S	N	S	N	S	N
C2	Estructura Lineal	S	S	N	N	S	S	N	N
C3	Acceso dinámico	S	S	S	S	N	N	N	N
Tablas con arreglos									
Listas con arreglos									
Arboles con arreglos									
Conjuntos con arreglos									
Tablas con listas ligadas									
Listas ligadas									
Arboles con listas ligadas									
Conjuntos con listas ligadas									

Fig 5.4 c). Malla adquirida para seleccionar TDH's con ordenamiento logico.

**Ordenado Logicamente**

**E6: Listas Ligadas**

¿Existen eliminaciones e inserciones muy frecuentes?

Respuesta : Si.

Sugerencia: Listas Doblemente Ligadas.

**E4, E8: Conjuntos**

¿Se permite que exista más de un elemento con el mismo valor?

Respuesta : Si

Sugerencia: Bolsas (Bags).

Respuesta : No.

Sugerencia: Conjuntos.

E3, E7: Arboles.

a) ¿La información deba reorganizarse con frecuencia?

Respuesta : Si.  
Sugerencia: Arboles AVL.

b) ¿Se requieren operaciones muy rápidas?

Respuesta : Si.  
Sugerencia: Arboles B

c) Arboles Heap.

E1, E5: Tablas.

a) Tablas con Hash.

b) ¿Se requieren operaciones muy rápidas, y se tienen mas de 100 registros?

Respuesta : Si.  
Sugerencia: Manejo con arboles B.

Archivos

a) ¿Se requieren accesos muy rápidos?

Respuesta : Si.  
Sugerencia: Hash.

b) ¿Se requieren accesos rápidos, y se tienen mas de 100 registros?

Respuesta : Si.  
Sugerencia: Manejo con arboles B.

c) ¿Los accesos no necesariamente deben ser rápidos y son pocos registros?

Respuesta : Si.  
Sugerencia: Archivos Secuenciales.

Durante todo el proceso de la aplicación, fui asesorada por la Dra. Hanna Oktaba de quien recibí una explicación detallada del Sistema Experto, asesoramiento durante cada uno de los pasos de la aplicación, así como valiosa información sobre TDA's que me permitió llegar a los resultados presentados.

#### Entrevista con la Dra Hanna Oktaba.

La entrevista con la Dra. Oktaba la realicé con la finalidad de especificar algunos conceptos que no estaban claros y para reafirmar que la información y resultados obtenidos hasta el momento eran correctos.

Esta entrevista se realizó como se describe a continuación:

Pregunta: ¿Como guiaría usted a un usuario no experto en la selección de un TDA adecuado?

Respuesta: A lo que la Dra. Oktaba respondió trataría de guiar al usuario utilizando una clasificación que enfocara más a las estructura de datos como estructuras algebraicas que como algoritmos, que es la clasificación con la que nacieron los TDA's. De esta manera no solo se guiaría a un usuario no experto hacia un TDA adecuado, sino también se le permitiría aprender este nuevo enfoque en caso de haber recibido una educación tradicional sobre Estructuras de Datos, es decir vistos como estructuras nacidas para soportar la funcionalidad de un algoritmo. Recalcó que el enfoque de los TDA's vistos como estructuras algebraicas independientes a los algoritmos es relativamente nuevo y que no existe una clasificación completamente establecida al respecto, por lo cual los resultados que yo había obtenido en esta investigación eran un aproximación válida.

Pregunta: ¿Porqué surgió este cambio en el punto de vista de los TDA's?

Respuesta: A lo que respondió que antes se construían algoritmos para realizar alguna tarea específica y las Estructuras de Datos surgieron bajo la necesidad de soportar estos algoritmos, por lo tanto cuando se enseñaban Estructuras de Datos, únicamente se mostraban los algoritmos que

permitían manipularlos y bajo ejemplos específicos. Entonces nació la necesidad de contar con un nivel de abstracción en los TDA's, de manera que pudiesen ser estudiados como estructuras independientes de los algoritmos y de esta manera obtener una clasificación independiente del lenguaje y la representación.

Pregunta: ¿Cree que existe una clasificación estándar en los TDA's? Cual sería esta?

Respuesta: Ante esta pregunta, la Dra. Oktaba respondió que en varios libros había encontrado diferentes clasificaciones, todas perfectamente válidas ya que ninguna violaba ningún axioma de un TDA y respetaban un orden lógico. Mencionó que básicamente en la bibliografía había encontrado que los TDA's estaban clasificados en: Estructuras Lineales, Estructuras Jerárquicas, Conjuntos, Gráficas y Diccionarios, y que en muchos de ellos la diferencia principal radicaba en la terminología utilizada, por lo cual era importante establecer un diccionario de equivalencias terminológicas.

Pregunta: ¿Cree que con los criterios de :

- Tipo de acceso (secuencial, aleatorio)
- Tipo de orden en las operaciones (ordenado lógicamente, ordenado temporalmente)
- Tipo de estructura (lineal, jerárquica)
- Estructura acotada

se podría elegir un TDA específico?

Respuesta: A lo que respondió que estos criterios le parecían aceptables y que de momento no se le ocurría ningún otro, pero aclaró que era importante que estos fuesen ortogonales y no criterios que de alguna manera se ligaran.

Pregunta: ¿Las características de un TDA las definen sus axiomas?

Respuesta: A lo que respondió afirmativamente y explicó que los axiomas definían lo que era el TDA en sí y como funcionaba.

**Pregunta:** ¿Que es un anillo y porque no esta en algunos libros como un TDA independiente?

**Respuesta:** Ante esta pregunta la Dra. respondió que podría verse un anillo como un buffer circular o como una cola acotada, esto dependía de la aplicación que se le diese. Mencionó que este era un ejemplo de la necesidad de estandarizar la terminología y finalmente aclaró que en algunos libros no contemplaban los anillos como un tipo de estructura independiente por estar cubierto dentro del contexto de las colas acotadas, ya que al implementar un algoritmo para el manejo de una cola acotada, se presentaba la necesidad de implementarlo como una cola circular. Adicionalmente mencionó que esto sucedía porque no existía aún un estándar que definiera claramente la ubicación de estos TDA's y que es un tema que todavía está en etapa de investigación.

**Pregunta:** Encontré que en la literatura un poco más antigua, no se consideran a las tablas como TDA's, sino como algoritmos, por ejemplo el algoritmo con tablas Hash ¿ Cual es el enfoque nuevo sobre las tablas?

**Respuesta:** A lo que la Dra. Oktaba respondió que ella veía a las tablas con las propiedades de un diccionario, es decir una estructura que permitía almacenar registros con información, la cual podría ser accesada y actualizada con el uso de una llave que la identificara. También mencionó que se podía ver las tablas desde diferentes puntos de vista, por ejemplo si se las veía con el enfoque de un implementador, a el le interesaría verlas como tablas hash y si se utilizaba el enfoque de un usuario se las podría ver como estructuras de almacenamiento y acceso de información y no le interesaría como fueron implementadas.

Adicionalmente mencionó que debido a la falta de estandarización en la clasificación de los TDA's, las tablas podrían estar dentro de la clasificación de estructuras lineales si se las veía como arreglos y dentro de la clasificación de estructuras jerárquicas si se las veía como diccionarios, pues el acceso por llaves de una tabla a otra proporcionaba un manejo similar al de un árbol.



Al finalizar la entrevista con la Dra. Oktaba, enfatizó que los resultados que yo había obtenido, eran a su parecer una aproximación válida, aunque era posible que otro experto tuviese un parecer contrario a mi propuesta.

#### IV. Conclusiones.

La conclusión a la que llegué es que el área de Tipos de Datos Abstractos es bastante extensa, y que para poder hacer una aportación significativa es necesario hacer una investigación mucho mas profunda que la expuesta en este documento.

Deseo aclarar que la intención de esta investigación fue aplicar algunas de las técnicas de Adquisición del Conocimiento expuestas en esta tesis y no así realizar una investigación profunda sobre TDA's.

Es importante mencionar que me hubiese gustado explotar mucho más las técnicas mencionadas en este documento, así como seguir lo más apegadamente posible las indicaciones que sugiere la técnica de entrevista. Esto no fue posible debido a la corta duración de la mayoría de estas. Por otra parte pienso que al utilizar la técnica de "Malla de Repertorio", cubrí los objetivos que esta propone, que son los de organizar la información obtenida de las entrevistas con los expertos y hacer más claros los conceptos obtenidos en estas.

## BIBLIOGRAFIA Y REFERENCIAS

- [1] SACERDOTI EARL D. "Managing Expert-System Development".  
AI Expert (Mayo 1991) (26-33).
- [2] SRA Computer Training Library. "Fundamentals of  
Knowledge Based System".
- [3] HART Anna. "Knowledge Elicitation: issues and methods".  
CAD. Vol 17 No. 9 (Nov 1985) (455-462).
- [4] MACDONALD B. A. y WITTEN I. H. "A framework for  
Knowledge Acquisition through Techniques of  
Concept Learning".  
IEEE Vol.19 No.3 (Mayo-Junio 1989) (499-512).
- [5] FRIEDLAND P. "Acquisition of Procedural Knowledge from  
Domain Experts".  
IEE Principles of Expert Systems.  
International Joint Conference on  
Artificial Intelligence. (1981) (856-861).
- [6] KOBAYASHI S. y KOTARO N. "Knowledgege Compilation and  
Refinement for Fault Diagnosis".  
IEEE Expert. Knowledge Acquisition.  
(Octubre 1991) (39-46).
- [7] KAWAGUCHI A., MOTODA H. y MIZOGUCHI H. R. "Interview-  
Based Knowledge Acquisition Using Dynamic  
Analysis".  
IEEE Expert. Knowledge Acquisition.  
(Octubre 1991) (47-60).
- [8] CLANCEY W. J., BARBANSON M. "Using the System-Model-  
Operartor Metaphor for Knowledge  
Acquisition".  
IEEE Expert. Knowledge Acquisition.  
(Octubre 1991) (61-65).
- [9] KAWAGUCHI A., MIZOGUCHI R., YAMAGUCHI T. y KAKUSHO O.  
"SIS: A Shell for Interview Systems".  
Knowledge Acquisition.
- [10] GAINES B. R. "An Overview of Knowledge-Acquisition and  
Transfer".  
Knowledge-Based Systems Vol. 1.  
Academic Press Ltd. (1988) (5-22).

- [11] BYLANDER T. y CHANDRASEKARAN B. "Generic Tasks for Knowledge-based Reasoning: the 'right' Level of Abstraction for Knowledge Acquisition". Knowledge-Based Systems Vol. 1. Academic Press Ltd. (1988) (65-77).
- [12] LAFRANCE M. "The Knowledge Acquisition Grid: a Method for Training Knowledge Engineers". Knowledge-Based Systems Vol. 1. Academic Press Ltd. (1988) (81-91).
- [13] LITTMAN D. C. "Modelling Human Expertise in Knowledge Engineering: some preliminary observations". Knowledge-Based Systems Vol. 1. Academic Press Ltd. (1988) (93-104).
- [14] SCOTT C. A., CLAYTON J. y GIBSON E. "A Practical Guide to Knowledge Acquisition". Addison-Wesley. (1991).
- [15] MITTAL S. y DYM C. "Knowledge Acquisition from Multiple Experts". A. I. Magazine Vol. 6. (1985) (32-36)
- [16] HAWKINS D. "An Analysis of Expert Thinking". Int. J. Man-Machine Studies 18 (1983) (1-47).
- [17] MARTINEZ A. M., "El aprendizaje en un Sistema Experto". Conferencia MEXICON 88, Guad-Mex, (1988).
- [18] OBREGON A. "Inteligencia Artificial y Sistemas Expertos" Centro Internacional de Soporte para America Latina. IBM de Mexico. (Noviembre 1990).
- [19] AUSSENAC N. "Acquisition des Connaissances". Bulletin de L'AFIA. Número 5. Marzo 1991 (15-33).
- [20] PATTERSON D. W. "Introduction to Artificial Intelligence and Expert Systems". Ed. Prentice-Hall (1991).
- [21] KAHN G., NOWLAN S. y MCDERMOTT J. "Strategies for Knowledge Acquisition". IEEE Trans. Vol. PAMI-7, No. 5 (Sept 1985) (511-522).
- [22] KOLOKOURIS A. T. "Machine Learning". Byte. (nov 1986) (225-231).

- [23] HARRISON R. "Abstract Data type in MODULA-2".  
Jhon Wiley & Sons. 1898.
- [24] AHO A., HOPCROFT J., ULLMAN J. "Data Structures and Algorithms".  
Ed. Addison-Wesley. (1987).
- [25] BOOCH G., "Software components with Ada"  
The Benjamin/Cummings Publishing Company,  
Inc. (1986).
- [26] GAINES B. R. "Knowledge Aquisition Systems".  
Knowledge-Engineering Vol. 1. "Fundamentals"  
Ed. Hojjat Adeli. Mc. Graw Hill. (1990) .

## APENDICE A

### HERRAMIENTAS DE SOFTWARE PARA EL APOYO DE LA ADQUISICION DEL CONOCIMIENTO

Como se mencionó con anterioridad uno de los problemas mayores, que explica en parte los altos costos y el fracaso en algunas experiencias del desarrollo de los Sistemas Expertos, es debido al problema de la Adquisición del Conocimiento. Es por esta razón que se empezaron a realizar investigaciones sobre herramientas automatizadas que ayudan parcial o totalmente al Ing. del Conocimiento en esta labor.

Los primeros trabajos en este dominio datan de 1979 con el sistema TEIRESIAS (de Davis y Lenat). Esta herramienta fue diseñada para proveer asistencia en la construcción de Sistemas Expertos. Facilita la transferencia interactiva del conocimiento de un experto al sistema, en un dialogo de alto nivel conducido en un subconjunto restringido del Lenguaje Natural. Con TEIRESIAS la Adquisición del Conocimiento se hace a partir de un dialogo entre el experto y el sistema y su apoyo sobre los modelos de reglas contenidos en el sistema MYCIN (Davis 1979). Trata de enriquecer la base misma al tiempo que la valida.

Desde entonces a la fecha se han investigado y desarrollado una gran cantidad de herramientas para la Adquisición del Conocimiento. A continuación se proporciona una breve clasificación de estas en base al tipo de aprendizaje y transferencia de información:

Los sistemas de generación de conocimientos implementan la Adquisición del Conocimiento por medio de la inducción de modelos a partir de la experiencia sin contar con el soporte cultural. Ejemplos: ATOM (Gaines, 1977) y AM (Davis & Lenat, 1982).

- Los sistemas de modelado de la expertez implementan la Adquisición del Conocimiento a través de la mímica del comportamiento del experto. Ejemplos: INDUCE (Michalsky & Chilausky, 1980).

- Los sistemas de reforzamiento de la ejecución implementan la Adquisición del Conocimiento a partir de la ejecución de la retroalimentación. Ejemplos: PERCEPTRON (Rosenblatt, 1958) y STeILA (Gaines & Andrae, 1966).

- Los sistemas de obtención del conocimiento implementan la Adquisición del Conocimiento a través de entrevistas al experto. Ejemplos: ETS (Boose, 1985), KITTEN (Gaines & Shaw, 1986), MORE (Kahn, Nowlan & McDermott, 1985) y SALT (Marcus, McDermott & Wang, 1985).

- Los sistemas de estructuración del conocimiento implementan la Adquisición del Conocimiento por analogía. Ejemplos: TEIRESIAS (Davis & Lenat, 1982), CYC (Lenat, Prakash & Shepherd, 1986).

- Los sistemas de leyes básicas implementan la Adquisición del Conocimiento por medio de la construcción de modelos físicos a través de la simulación de lenguajes como SIMULA (Nygaard & Dahl, 1981).

- Los sistemas de principios sistémicos implementan la Adquisición del Conocimiento por medio de la derivación de principios abstractos como se ejemplifica con el lenguaje categórico-teórico OBJ (Goguen & Meseguer, 1983).

Esta misma taxonomía se muestra en una tabla en la Fig. A.1

Tipos de sistemas	Técnicas de aprendizaje	Transferencia de Conocimiento		Ejemplos
		Entrada	Salida	
Generación de Conocimientos	Inducción	Experiencias	Modelos	ATOM AM
Modelado de la expertez	Mímica	Ejecución	Reglas	INDUCE
Reforzamiento de la ejecución	Aprendizaje	Retroalimentación	Políticas	PERCEPTION STELLA
Adquisición del conocimiento	Expresión	Expertez	Marcos de trabajo	ETS, MITTEN SALT, NORE
Estructuración del conocimiento	Analogía	Modelos	Analogías	TEIRESIAS CVC
Sistemas de Leyes Básicas	Simulación	Leyes	Modelos	SIMULA
Principios Sistémicos	Teoría de Categorías	Principios	Perspectivas	OBJ

Fig A.1 Tabla de Clasificación de Herramientas para la Adquisición del Conocimiento

Las herramientas actuales para la construcción de los Sistemas Expertos, están diseñadas para tareas específicas tales como diagnóstico y diseño. De acuerdo a esto, muchos sistemas para la Adquisición de Conocimientos están diseñados para adquirir el conocimiento para una clase particular de problemas. Estos utilizan la estructura de una tarea para guiar la Adquisición del Conocimiento. A continuación se describen algunas de tales herramientas.

## I. SIS. Un metasisistema de entrevistas. [7] [9]

SIS (Shell for Interview Systems) es un sistema prototipo a partir del cual se pueden generar sistemas de entrevistas para una aplicación específica.

A través de la investigaciones, se pudo apreciar que una entrevista por si misma forma una clase de tarea y que el entrevistador no solo posee conocimientos acerca de la tarea, sino también sobre la entrevista misma, lo cual hace del entrevistador un entrevistador experto.

A partir de tales investigaciones, se puede observar que la entrevista al igual que el diagnostico o el diseño es una clase de tarea, y que por lo tanto se puede representar su estructura con un conjunto de primitivas que son independientes tanto de las tareas del dominio como de la aplicación y solo son dependientes de la tarea de la entrevista. Estas primitivas pueden facilitar la construcción de sistemas de entrevistas, por lo que el objetivo principal era identificar tal conjunto de primitivas.

Esta ideas se han utilizado para generar SIS. Este metasisistema sirve como un esqueleto de un sistema de entrevistas. Provee las primitivas específicas de una tarea de entrevista y un conjunto de descriptores para representar el conocimiento de una entrevista para una tarea específica.

SIS implementa la Adquisición del Conocimiento utilizando análisis estático. Especifica las estrategias de preguntas al analizar a priori la naturaleza de la tarea, al margen del tipo de resolución del problema actual, es decir, el sistema únicamente considera los atributos estáticos de la Base de Conocimientos en lugar de los comportamientos resultantes de la utilización de la Base de Conocimientos.

Los sistemas de entrevistas generados por SIS, no contienen un proceso resolutor de problemas, es decir un componente que evalúe la ejecución del sistema para el cual es adquirido el conocimiento.

Un sistema de entrevistas generado por SIS comienza requiriendo información inicial para comenzar una entrevista y construir un modelo incompleto del dominio en forma de red. Este analiza una oración del experto y genera un conjunto de



"atenciones" de una tarea específica. Basado en tales atenciones, el sistema formula las preguntas para adquirir conocimiento nuevo. El modelo del dominio es refinado a través de ciclos respuesta/atención/pregunta.

#### **Modelo del dominio.**

SIS representa el modelo del dominio de la tarea para la aplicación como una red de nodos y ligas, es decir, una gráfica dirigida. Los nodos representan conceptos y la ligas relaciones entre los conceptos. Tanto los nodos como las ligas son representados como marcos con celdas, y cada celda tiene un conjunto de valores. Esta representación incluye características estándares tales como una jerarquía de clases y relaciones clase-instancia.

#### **Atenciones Genéricas.**

Las primitivas de una tarea específica de SIS, llamadas atenciones genéricas, son datos o pistas encontrados durante el curso de la entrevista, estas guían el flujo de la entrevista previniendo preguntas innecesarias y adquiriendo conocimiento nuevo de manera más eficiente. "Genérico" significa que estas atenciones son independientes del dominio e independientes de la tarea. Son entidades abstractas que deben ser instanciadas a una situación específica.

SIS cuenta con ocho primitivas básicas para la comparación, a las que llama atenciones primitivas.

#### **Implementación.**

El nivel superior de un sistema implementado es el nivel del sistema de entrevistas. Abajo de este existen dos niveles más. El nivel medio es el de la estructura interna, representa el sistema en términos de los módulos de programa específicos de SIS. Abajo, el nivel del conocimiento, especifica el comportamiento de cada módulo en el nivel de la estructura interna al describir declarativamente estos comportamientos. El conocimiento que se proporciona aquí, eventualmente determina el comportamiento del sistema de entrevistas.

### **Estrategia de entrevistas.**

La generación y el procesamiento de las atenciones son esenciales para la Adquisición del Conocimiento durante la entrevista en SIS. El sistema ejecuta cada pregunta y responde generando una atención correspondiente y procesándola. Cada atención tiene su propio conocimiento para realizar una pregunta al usuario. Los detalles del comportamiento de los que depende cada sistema de entrevistas son definidos por el conocimiento proporcionado en cada módulo del nivel de conocimiento.

### **Aplicaciones.**

Existen varios sistemas de entrevistas generados con SIS. Dos de ellos son SIS-More y IIS-LD. Estos son diferentes en los siguientes aspectos. Primero, las tareas de SIS-More son de diagnóstico, mientras que las de IIS-LD son de diseño. Segundo, los sistemas tienen diferentes niveles de Adquisición del Conocimiento. SIS-More extiende el modelo de la tarea, mientras que IIS-LD lo crea. El modelo de la tarea en SIS-More es dado como un conjunto de estrategias de entrevista y como la estructura del modelo del dominio, los cuáles son la entrada de SIS. SIS-More adquiere instancias del conocimiento de diagnóstico para una aplicación específica (por ejemplo el dominio del drenado de fluidos). En IIS-LD el modelo de la tarea no es una entrada para SIS, en su lugar, debe ser adquirido. La entrada de SIS es un tipo de meta-conocimiento para adquirir tal modelo. IIS-LD no adquiere datos de instancias.

### **II. SIS-More. [7] [9] [21]**

SIS-More es un sistema de Adquisición de Conocimientos para la construcción de Sistemas Expertos de diagnóstico, cuyo modelo del dominio es una red de relaciones causales entre los síntomas y las hipótesis.

La tarea de SIS-More es la clasificación heurística. Este sistema adquiere conocimiento de diagnóstico en forma de reglas que relacionan síntomas a posible hipótesis. Las 8 preguntas estratégicas en More están basadas en las ideas o suposiciones hechas apriori, de manera que la resolución de problemas actuales no es necesaria. A esto último se le llama análisis estático.

La estrategia de diferenciación es utilizada para distinguir las hipótesis que aparecen para los mismos síntomas. A pesar de que esta estrategia es utilizada, los casos podrían originarse en dos hipótesis que son derivadas del mismo síntoma. Este tipo de problemas puede ser fácilmente predecido a priori y no requerir de un resolutor de problemas.

Las estrategias de preguntas en More confirman si el conocimiento adquirido hasta cierto punto, es suficiente para resolver el problema.

#### **Dialogo.**

El sistema nota las ambigüedades en las relaciones entre los síntomas iniciales y las hipótesis, y formula preguntas para diferenciar la hipótesis. Pregunta si los nuevos síntomas provistos pueden ser una causa del síntoma ya conocido y produce tres reglas de diagnóstico.

### **III. IIS-LD. [7] [9]**

IIS-LD es un sistema de entrevistas para la creación de diseños lógicos de Bases de Datos basándose en el modelo entidad-relación, por lo tanto su tarea es enumerar las entidades necesarias y suficientes en el dominio de interés y las relaciones (actividades) entre ellas. Las entidades y relaciones, aquí se presentan como conceptos abstractos.

#### **Estrategia de entrevista.**

Las nueve atenciones genéricas en SIS son suficientes para describir la Estrategia de entrevistas en SIS-More y en IIS-LD.

La tarea de alto nivel en IIS-LD es "Diseña-BD". El sistema primero crea un modelo inicial y lo convierte en una descripción entidad-relación. Posteriormente, revisa y actualiza el modelo del dominio hasta que no existan más preguntas ni atenciones.

IIS-LD utiliza un conjunto de estrategias de preguntas para predecir posibles problemas que pueden ocurrir cuando el modelo del dominio actual construye una Base de Datos.

#### **Diálogo.**

En el dialogo, el sistema utiliza la Estrategia de planeación. Por ejemplo, como el sistema no conoce el nombre "parte", formula una pregunta para localizarla en una jerarquía de nombre. Dado que en este contexto "parte" no se refiere a una parte específica, la respuesta es "no" y IIS-LD instancia este nombre. En este punto, una atención es generada al analizar el modelo, y los verbos candidatos son elegidos y desplegados. IIS-LD lista los posible atributos del nombre. El dialogo continúa y IIS-LD produce como salida el esquema conceptual diseñado al final.

En IIS-LD las atenciones son generadas no solo cuando algo es impropio, sino también cuando el conocimiento base puede hacer la adquisición más eficiente.

#### **IV. IIS-DB. (7)**

El análisis estático, como el utilizado en SIS-More y en IIS-LD, es más adecuado para casos donde la meta-estructura del conocimiento del dominio es bien conocida y entendida. El análisis estático no es tan útil para el diseño lógico de una Base de Datos por muchas razones algunas de las cuales son:

- Es difícil preparar apriori preguntas estratégicas suficientes.
- Los expertos del dominio a menudo no están posibilitados para responder en el momento.
- Algunos problemas pueden ser notados únicamente durante el uso.

### **El Diseño Lógico y el Análisis Dinámico.**

IIS-DB es una versión mejorada de IIS-LD, e incorpora análisis dinámico para lograrlo. IIS-DB incluye a IIS-LD como uno de sus componentes. Al entrevistar al usuario, IIS-DB adquiere el conocimiento del dominio, que mapea el mundo real en la estructura de la Base de Datos que el experto/usuario del dominio desea construir. IIS-DB utiliza IIS-LD para adquirir el conocimiento inicial y entonces lo refina haciendo que el usuario utilice la Base de Datos implementada.

Las preguntas formuladas por IIS-DB asumen el último modelo que el usuario tiene en mente. Las deficiencias en el conocimiento aparecen en el desacuerdo entre el modelo asumido por la pregunta y el modelo implementado. El sistema puede formular preguntas basadas en el desacuerdo y modificar el modelo de acuerdo a esto simulando el comportamiento de un diseño experto de Bases de Datos. Si se proporcionara una interface con el usuario en lenguaje natural, el sistema podría explorar la causa del desacuerdo sin imponer una carga innecesaria al usuario.

Este proceso de refinamiento es iterativo y va en ciclos de construcción/identificación-del-problema/reconstrucción. IIS-DB evalúa la Base de Datos prototipo para identificar problemas, analizarlos, y corregir el modelo del dominio. Reconstruye la Base de Datos basándose en su modelo correcto y repite la evaluación.

IIS-DB inicia una entrevista con la pregunta actual, como una pista, basándose en las modificaciones que son hechas al diseño lógico. Una vez que la pista es obtenida, no existe una distinción substancial entre el análisis estático y el análisis dinámico. Las estrategias y mecanismo son los mismos, por lo que se puede obtener el mismo diseño de Base de Datos si la pista obtenida dinámicamente, se conoce desde el principio.

### **Configuración.**

IIS-DB consiste de IIS-LD y otros módulos incluyendo la interface del sistema manejador de la Base de Datos, una Base de Conocimientos para la construcción de la Base de Datos y un supervisor. El supervisor monitorea el comportamiento de cada módulo.

IIS-DB traslada las preguntas en lenguaje natural en preguntas escritas en un Lenguaje de Manipulación de Datos. El módulo de interface transfiere estas preguntas al Sistema Manejador de Base de Datos y recibe los resultados buscados.

#### Adquisición del Conocimiento.

En la primera parte de la Adquisición del Conocimiento, el usuario debe responder a las preguntas de la entrevista sin utilizar una Base de Datos prototipo. La segunda parte es la que utiliza la Base de Datos (lo cual lo distingue de IIS-LD), es decir adquiere el conocimiento utilizando análisis dinámico. La utilización de la Base de Datos, es un buen estímulo para el usuario ya que surgen varios requerimientos o se aclaran únicamente después de su uso. Dado que IIS-DB ayuda a detectar estos requerimientos, el sistema gradualmente se mejora y eventualmente adquiere el siguiente conocimiento:

- Entidades (nombres) y sus atributos.
- Relaciones (verbos) y sus atributos.
- Ligas entre entidades y relaciones.
- Sinónimos entre entidades y partes de relaciones.
- Ligas entre dos relaciones y
- Sinónimos para relaciones.

La Adquisición del Conocimiento utilizando el análisis dinámico ofrece muchas ventajas. Primero, la Adquisición del Conocimiento utilizando el análisis estático, puede ser incompleta reduciendo el número de preguntas detalladas que pueden haber sido requeridas para hacer un modelo completo. Por otra parte, dado que no es necesario responder todas las preguntas mientras el conocimiento del usuario no esté bien organizado, la carga al usuario es reducida considerablemente.

Otra ventaja, es que el solucionador de problemas simula al usuario para producir nuevos requerimientos. Este es efectivo haciendo explícito el conocimiento ambiguo al poner al usuario en un medio ambiente de solución de problemas. Adicionalmente el sistema puede manejar un requerimiento que el usuario no haya pensado al tiempo del análisis estático.

## V. El proyecto ESPRIT. (19)

En esta sección se presentan los diferentes proyectos ESPRIT 1 y 2 en el dominio de la Adquisición del Conocimiento destacando la participación de los equipos franceses, y las actividades de investigación de tales grupos en este dominio realizadas en Francia.

### Los proyectos ESPRIT 1.

**ESB (Expert System Builder)**  
**Constructor de Sistemas Expertos.**

El proyecto ESB fue creado para la industrialización de Sistemas Expertos destinados para la utilización de usuarios no experimentados en Inteligencia Artificial. Este es comercializado en la forma de un medio ambiente completo que consta de los siguientes módulos:

- Un sistema orientado a objetos (FLAME).
- Un formalismo de la representación de conocimientos (CONCEPT).
- El mecanismo de inferencia. (BESB).
- La definición de un modelo conceptual y la interface hombre-máquina (MMI).

Este ha sido retomado por el proyecto ESPRIT 2 como PROMISE sobre la interface hombre-maquina.

**KADS (Knowledge Acquisition and Design Support)**  
**Una metodología para el desarrollo de Sistemas Expertos.**

KADS es una herramienta para la Adquisición del Conocimiento. Proporciona soporte en la transferencia de la tecnología de Sistemas Expertos a través del dominio comercial. Este posee una biblioteca de modelos de tareas genéticas, recopila modelos de interpretación y posee un módulo para la adquisición de estructuras del conocimiento.

Los proyectos ESPRIT 1 concernientes a la Adquisición del Conocimiento concluyeron con el éxito de KADS, proyecto ambicioso y vasto que ha producido resultados metodológicos fundamentales. La metodología propuesta por KADS se reafirma actualmente como el principal candidato a un estandar Europeo dentro del marco de la adaptación de métodos para la Ingeniería del Conocimiento.

## Los proyectos ESPRIT 2.

Existen tres proyectos dentro del ESPRIT 2 cuyo objetivo está relacionado con la Adquisición del Conocimiento, estos son KADS II, VITAL y ACKnowledge.

### ACKnowledge (Acquisition of Knowledge) Adquisición de Conocimiento.

Este proyecto tiene como objetivo la realización de un taller de Ingeniería del Conocimiento, que integra metodologías, técnicas y utilerías de Adquisición del Conocimiento y del aprendizaje automático para dar como resultado un medio ambiente coherente.

Se enfocan de manera especial los objetivos de mecanismos de aprendizaje, edición y visualización de la Base de Conocimientos.

### KADS II

Una metodología avanzada y comprensiva para la integración del desarrollo de Sistemas Expertos.

El proyecto KADS II pretende ofrecer una metodología que cubra completamente el ciclo de vida de un proyecto de Sistemas Expertos. Propone estándares para el control del proyecto y el desarrollo del sistema y establece una conexión entre los métodos de desarrollo de Sistemas Expertos y las metodologías de la Ingeniería del Conocimiento. La aproximación de KADS II está basada en KADSI, en la idea de un proceso iterativo de construcción metodológica. La construcción de un Sistema Experto está descrita con la ayuda de modelos con diferentes niveles de abstracción.

Este proyecto tiene como objetivo facilitar la especificación formal del modelo y permitir la reutilización de componentes, integrando los resultados recientes en las tareas genéricas y los métodos de resolución de problemas.



**VITAL.**

Una metodología basada en la mesa de trabajo para dar soporte al ciclo de vida de los Sistemas Expertos.

El objetivo global del proyecto es la realización de una plataforma lógica integrada por utilerías que cubran la unión del ciclo de vida de los Sistemas Expertos (construcción, mantenimiento, y manejo del proyecto). Se apoya, como punto de partida, en la evaluación y extensión de métodos existentes, técnicas y utilerías relevantes de la Ingeniería del Conocimiento (KEATS), el aprendizaje automático (CHARADE) y los medios ambientes gráficos (GKE KBS).

Las partes de adquisición y aprendizaje están puestas en marcha por ONERA.

En la parte relacionada con la Adquisición del Conocimiento y el aprendizaje simbólico, se trata de definir como se integran las diferentes metodologías y técnicas conocidas de obtención y adquisición de conocimientos en el marco global del ciclo de vida de los Sistemas Expertos para realizar e integrar las utilerías correspondientes.

**El proyecto ESPRIT 3.**

Al parecer la Comunidad de Estados Europeos no desea seguir financiando muchas de las nuevas investigaciones en Adquisición del Conocimiento. A pesar de ello está atendiendo la introducción de miles de utilerías para la industria como término de los proyectos ESPRIT I y II, a manera de medir el interés suscitado por estas utilerías y su eficacia.

## APENDICE B

### PROGRAMACION DE UN PROTOTIPO DE UN SISTEMA EXPERTO DE COMPONENTES REHUSABLES DE TIPOS DE DATOS ABSTRACTOS

#### I. Introducción.

Este apendice incluye los programas de un prototipo de un Sistema Experto de Componentes rehusables de Tipos de Datos Abstractos (TDA), mismo que se describió en el capítulo V.

Este sistema guia al usuario atravez de ventanas de menu hacia la seleccion adecuada del TDA. Así mismo le permite revisar los conceptos, especificaciones y aplicaciones del TDA elegido.

El lenguaje que se utilizó en la etapa de codificación del prototipo fue EIFFEL. Este lenguaje en Orientado a Objetos y fue escogido por adecuarse a la aplicacion ya que en esta cada TDA podia ser identificado como un clase, así como sus especificaciones, conceptos y aplicaciones.

Este prototipo esta intrumentado en una computadora SUN con Sistema Operativo UNIX en el Intituto de Investigación en Matemáticas Aplicadas y Sistemas (I.I.M.A.S.) de la U.N.A.M.

**II. Programas.**

**ARCHIVO: ANILLOS.E**

```
class ANILLOS export simbolo
inherit
  COMMTDA;
feature
  simbolo : STRING is "anillo";

  Create is
  do
    window_Create;
  end;

end;
```

**ARCHIVO: ARBOLES.E**

```
class ARBOLES export simbolo
inherit
  COMMTDA;
feature
  simbolo : STRING is "arbol";

  Create is
  do
    window_Create;
  end;

end;
```

**ARCHIVO: AXIOMA.E**

```
class AXIOMA export t
inherit
  COMANDOS;
  MAP_ELEMENT;
feature
  Create is
  do
    window_create;
  end;
```

```

ejecuta(simb : STRING) is
local faxi : FILE;
    nombre: STRING;

do
    -- crea la ventana donde se desplegaran los axiomas de tda
elegido
attach(t);
set_height(on_device.height - 6);
set_width(on_device.width - 10);
put_at(0,0);
set_frame;
textual.Create(1000);
nombre.Creates(20);
-- Abre y lee el archivo donde esta el texto que describe los
axiomas
nombre.append(simb.duplicate);
nombre.append(".axi");
faxi.Create(nombre);
faxi.open_read;
from
    faxi.readline;
until
    faxi.end_of_file
loop
    textual.append(faxi.laststring.duplicate);
    textual.append("\n");
    faxi.readline;
end; -- loop
faxi.close;
set_priority (1);
display;
io.readchar;
remove_text;
end;
end; -- axioma

```

#### ARCHIVO: AYUDA.E

```

class AYUDA export nivel
inherit
    COMMAND;
    MAP_ELEMENT
feature
    nivel :INTEGER;

```

```

Create is
do
  window_create;
end;

ejecuta is
local faxi : FILE;
  nombre: STRING;
  alto, limite, i : INTEGER;
  cnum : CHARACTER;
do
  -- crea la ventana donde se desplegaran los axiomas de tda
elegido
  attach(target);

  inspect
    nivel
  when 1 then
    alto:= on_device.height - 3;
    limite:= 7;
  when 2,3
    alto:= on_device.height - 5;
    limite := 3;
  when 4,5,6,7
    alto:= on_device.height - 5;
    limite := 7;
  end; -- inspect

  set_height(alto);
  set_width(on_device.width - 10);
  put_at(0,0);
  set_frame;
  from
    textual.Create(1000);
    nombre.Create(20);
    i:= nivel;
  until
    i > limite
  loop
    -- Abre y lee el archivo donde esta el texto que describe los
axiomas
    nombre.append("ayuda");
    cnum.et_code(nivel + 48);
    nombre.append(cnum);
    nombre.append(".doc");
    faxi.Create(nombre);
    faxi.open_read;
    from
    faxi.readline;
    until
    faxi.end_of_file

```

```

loop
textual.append(faxi.laststring.duplicate);
textual.append("\n");
faxi.readline;
end; -- loop
faxi.close;
set_priority (1);
display;
io.readchar;
remove_text;
end; --for
end;

end; -- ayuda

```

**ARCHIVO: COLAS.E**

```

class COLAS export simbolo

inherit
  COMMTDA;

feature
  simbolo : STRING is "cola";

  Create is
  do
    window_Create;
  end;

end;

```

**ARCHIVO: COMADOS.E**

```

deferred class COMANDOS export
  set_t, t, ejecuta
feature
  t: COMMTDA;

  set_t (map: COMMTDA) is
    -- Liga el comando actual a la ventana
    do
      t := map
    end; -- set_t

  ejecuta (simb : STRING) is
    deferred
    end; -- execute
end -- class COMANDOS

```

**ARCHIVO: COMMAND.E**

```

deferred class COMMAND export
  set_target, target, execute
feature

  target: MENUS;

  set_target (map: MENUS) is
    -- Liga el comando atual a la ventana.
    do
      target := map
    end; -- set_target

  execute is
    deferred
    end; -- execute

end -- class COMMAND

```

**ARCHIVO: COMMTDA.E**

```

class COMMTDA export
  execute, target, has_frame, key_stroke

inherit
  COMMAND;
  MAP_ELEMENT
feature

  simbolo : STRING is
    deferred
    end;

  key_stroke : COMANDOS;

  -- Ejecuta el comando y abre la ventana del menu donde se
  -- seleccionara el aspecto que se desea revisar del TDA
  execute is
    do
      attach(target);
      set_height(on_device.height - 6);
      set_width(on_device.width - 10);
      put_at(2,2);
      set_frame;
      menu;
      ciclos;
    end; -- execute;

```



```

menu_operacion : POPUP_MENU[COMANDOS];

-- Crea el menu
menu is
  local
    concep,algorit : CONCEPTO;
    axioma : AXIOMA;
    fin :FIN

  do
    concep.Create; axioma.Create; algorit.Create;
    fin.Create;
    menu_operacion.Create(Current);
    menu_operacion.associate("Conceptos
Generales", concep);
    menu_operacion.associate("Especificaciones", axioma);
    menu_operacion.associate("Aplicaciones", algorit);
    menu_operacion.associate("Salir", fin);
  end;

-- Ciclo de lectura y ejecucion de un comando
ciclos is
  local
    last_choice,sale : COMANDOS;
    fin : FIN;

  do
    from
      fin.Create;
      sale:= fin;
      display;
      move_cursor (1, 1)
    until
      not key_stroke.Void and then
      key_stroke.equal(sale)
    loop
      menu_operacion.init(5,5);
      -- Main menu display
      menu_operacion.show;
      display;
      menu_operacion.hide;
      -- User's choice input
      from
        key_stroke := menu_operacion.user_answer
      until
        menu_operacion.submenu.Void
      loop
        key_stroke := menu_operacion.user_answer
      end;
      if not key_stroke.Void then

```

```

                                -- Command creation and
execution
                                last_choice.Clone (key_stroke);
                                last_choice.set_t (Current);
                                last_choice.ejecuta(simbolo);
                                end;
                                display;
                                remove_text;
                                hide
                                end
                                end; -- ciclo

end;

```

## ARCHIVO: CONCEPTO.E

```

class CONCEPTO export t
inherit
    COMANDOS;
    MAP_ELEMENT
feature
    Create is
    do
        window_create;
    end;

    ejecuta(simb : STRING) is
    local faxi : FILE;
        nombre: STRING;

    do
        -- Abre la ventana donde se desplegara los conceptos del tda
    elegido
        attach(t);
        set_height(on_device.height - 6);
        set_width(on_device.width - 10);
        put_at(0,0);
        set_frame;
        textual.Create(1000);
        -- Abre y lee el archivo donde leera el texto que describe a
    los conceptos
        nombre.Create(20);
        nombre.append(simb.duplicate);
        nombre.append(".con");
        faxi.Create(nombre);
        faxi.open_read;
        from
            faxi.readline;

```

```

until
    faxi.end_of_file
loop
    textual.append(faxi.laststring.duplicate);
    textual.append("\n");
    faxi.readline;

    end; -- loop
faxi.close;
set_priority (1);
display;
io.readchar;
remove_text;
end;

end; -- concepto

```

**ARCHIVO: CONJUNTOS.E**

```

class CONJUNTOS export simbolo
inherit
    COMMDA;

feature
    simbolo : STRING is "conjunto";

    Create is
    do
        window_Create;
    end;

end;

```

**ARCHIVO: FIN.E**

```

class FIN export
    set_t, ejecuta
inherit
    COMANDOS
feature
    ejecuta(sim : STRING) is
        -- Nothing to do in this case.
        do
            end; -- execute
end -- class FIN

```

## ARCHIVO: LISTAS.E

```
class LISTAS export simbolo, execute, set_target, target
inherit
    COMMTDA;

feature
    simbolo : STRING is "lista";

    Create is
    do
        window_Create;
    end;

end;
```

## ARCHIVO: LOGICOS.E

```
class LOGICOS export
    execute, target, has_frame

inherit
    COMMAND;
    MAP_ELEMENT;

feature
    jer, lin, din : STRING;
    tabla : TABLAS;
    lista : LISTAS;
    arbol : ARBOLES;
    conjunto : CONJUNTOS;

    -- Abre y despliega la ventana donde se realiza el
    cuestionario
    -- que selecciona el tda adecuado
    execute is

    do
        tabla.Create;
        lista.Create;
        arbol.Create;
        attach(target);
        set_height(15);
        set_width( 50);
        put_at(2,2);
        set_priority(1);
        display;
    menu;
    ciclos;
```

```
display;
seleccion;
end; -- execute;

-- Crea el menu del cuestionario
menu is
do
    textual.Create(500);
    textual.append(" ORDENAMIENTO LOGICO \n\n");
    textual.append("- Estructura jerarquica (s/n) _ \n");
    textual.append("- Estructura lineal (s/n) _ \n");
    textual.append("- Acceso dinamico (s/n) _ \n");
    show;
    display;
    set_frame
end; -- menu

-- Preguntas del cuestionario
ciclos is
do
    move_cursor(4,32);
    echo;
    display;
    from
        jer.Create(1);
    until jer.equal("s") or jer.equal("n")
    loop
        read_string(1);
        jer.append(last_string);
        if not (jer.equal("s") or jer.equal("n") ) then
            bell;
            move_cursor(4,32);
        end; -- if
    end; --loop

    move_cursor(5,32);
    echo;
    display;
    from
        lin.Create(1);
    until lin.equal("s") or lin.equal("n")
    loop
        read_string(1);
        lin.append(last_string);
        if not (lin.equal("s") or lin.equal("n") ) then
            bell;
            move_cursor(5,32);
        end; -- if
    end; --loop

    move_cursor(6,32);
```

```

echo;
display;
from
  din.Create(1);
until din.equal("s") or din.equal("n")
loop
  read_string(1);
  din.append(last_string);
  if not (din.equal("s") or din.equal("n") ) then
    bell;
    move_cursor(6,32);
  end; -- if
end; --loop

noecho;
end; -- ciclo

-- Selecciona el tda adecuado
seleccion is
  local ventanita : MENUS;
      leyenda      : STRING;
      com          : COMMAND;
do
  ventanita.Create;
  ventanita.set_height(4);
  ventanita.set_width( 40);
  ventanita.put_at(11,10);
  leyenda.Create(100);
  leyenda.append("SELECCION ADECUADA : \n");
  if jer.equal("s") and lin.equal("s") then
    -- Tablas
    com.clone(tabla);
    if din.equal("n") then
      leyenda.append("\nTABLAS IMPLEMENTADAS CON LISTAS
LIGADAS \n");
    else
      leyenda.append("\nTABLAS IMPLEMENTADAS CON ARREGLOS
\n");
    end; --if
  end; --if
  if jer.equal("n") and lin.equal("s") then
    -- Listas
    com.clone(lista);
    if din.equal("n") then
      leyenda.append("\nLISTAS IMPLEMENTADAS CON LISTAS
LIGADAS \n");
    else
      leyenda.append("\nLISTAS IMPLEMENTADAS CON ARREGLOS
\n");
    end; --if
  end; --if
  if jer.equal("s") and lin.equal("n") then
    -- Arboles

```

```

com.clone(arbol);
if din.equal("n") then
leyenda.append("\nARBOLES IMPLEMENTADOS CON LISTAS
LIGADAS \n");
else
leyenda.append("\nARBOLES IMPLEMENTADOS CON ARREGLOS
\n");
end; --if
end; --if
if jer.equal("n") and lin.equal("n") then
-- Conjuntos
com.clone(conjunto);
if din.equal("n") then
leyenda.append("\nCONJUNTOS IMPLEMENTADOS CON LISTAS
LIGADAS \n");
else
leyenda.append("\nCONJUNTOS IMPLEMENTADOS CON ARREGLOS
\n");
end; --if
end; --if
ventanita.priority(1);
ventanita.associate_text(leyenda);
ventanita.display;
ventanita.set_frame;
display;
show;
io.readchar;
ventanita.put_at(2,2);
ventanita.associate_text(" ");
ventanita.hide;
com.set_target(ventanita);
com.execute;
remove_text;
hide;
unset_frame;

end; -- seleccion

end

```

**ARCHIVO: MENUS.E**

```

class MENUS export
choice_menu, set_up,
display, visible, xpos, ypos,
subwindow_position, subwindow_go,
show, hide, priority, associate_text,
subwindow_forth, subwindow_start, subwindow_offright,
subwindow,
height, width, attach, put_at, set_reverse,
user_xposition, user_yposition,
set_frame, unset_frame, set_height, set_width,
on_device, has_frame, cursor_on,

```

```

    move_cursor, re_init, prepare_choice_menu
inherit
MAP_ELEMENT
    rename
        hor_cursor as user_xposition,
        vert_cursor as user_yposition;

feature

    Create is
    do
        window_Create
    end; -- Create

    set_up is
    do
        -- Organiza la jeraquia de las ventanas de menus.
        set_width (on_device.width - 6);
        set_height (on_device.height - 3);
        put_at (2, 2);
        set_frame;
        cursor_enable;
        arch_menu;
        prog_menu;
        selec_menu;
        guia_menu;
        general_menu
    ensure
        width = on_device.width - 6;
        height = on_device.height - 3;
        has_frame;
        cursor_on;
        not choice_menu.Void
    end; -- set_up

    choice_menu, menu_guia, menu_select, prog, arch: POPUP_MENU
[COMMAND];

-- Menu para elegir el tda adecuado para la manipulacion de
archivos
arch_menu is
local
    nada: QUIT;
do
    nada.Create;
    arch.Create (Current);
    arch.associate ("Menu Inhabilitado", nada);
end; -- arch_menu

```



```
-- Menu para elegir el tda adecuado para la manipulacion de
estructuras
-- temporales
prog_menu is
  local
    logico : LOGICOS;
    temporal :TEMPORALES;
  do
    prog.Create(Current);
    logico.Create; temporal.Create;
    prog.associate("Operaciones ordenadas logicamente",
logico);
    prog.associate("Operaciones ordenadas temporalmente",
temporal);
  end; -- prog_menu
```

```
-- Menu para elegir un tda especifico
selec_menu is
  -- menu de guia para la seleccion de un TDA
  local
    quit_state: QUIT;
    lista: LISTAS;
    cola : COLAS;
    pila : PILAS;
    anillo : ANILLOS;
    arbol: ARBOLES;
    tabla : TABLAS;
    conjunto : CONJUNTOS;

  do
    quit_state.Create; lista.Create; cola.Create;
    arbol.Create; pila.Create; anillo.Create;
    tabla.Create; conjunto.Create;
    menu_select.Create(Current);
    menu_select.associate("Listas", lista);
    menu_select.associate("Colas", cola);
    menu_select.associate("Pilas", pila);
    menu_select.associate("Anillos", anillo);
    menu_select.associate("Arboles", arbol);
    menu_select.associate("Tablas", tabla);
    menu_select.associate("Conjuntos", conjunto);

  end; --selec_menu;
```

```
-- Menu para elegir el tipo de tda
guia_menu is
  local
    quit_state: QUIT;

  do
    quit_state.Create;
    menu_guia.Create(Current);
```

```

        menu_guia.associate_submenu("Tiempo de vida
permanente", arch);
        menu_guia.associate_submenu("Tiempo de vida de un
programa", prog);
        end; --guia_menu;

        -- Menu general
        general_menu is
        local
            quit_state: QUIT;
        do
            quit_state.Create;
            choice_menu.Create (Current);
            choice_menu.associate_submenu ("Seleccion de un TDA", menu_select);
            choice_menu.associate_submenu ("Guia para la seleccion de un TDA",
            menu_guia);

            choice_menu.associate ("Salir", quit_state)
        end; -- general_menu

        -- Prepara el menu
        prepare_choice_menu is
        do
            choice_menu.init (2,10);
            choice_menu.show
        end; -- prepare_choice_menu

    end -- class MENUS

```

## ARCHIVO: PILAS.E

```

class PILAS export simbolo

inherit
    COMMTDA;

feature
    simbolo : STRING is "pila";

    Create is
    do
        window_Create;
    end;

end;

```

**ARCHIVO: QUIT.E**

```

class QUIT export
  set_target, execute
inherit
  COMMAND
feature
  execute is
    -- Nothing to do in this case.
    do
    end; -- execute
end -- class QUIT

```

**ARCHIVO: TABLAS.E**

```

class TABLAS export simbolo
inherit
  COMMTDA;
feature
  simbolo : STRING is "tabla";

  Create is
  do
    window_Create;
  end;

end;

```

**ARCHIVO: TDA.E**

```

class TDA export
  map, key_stroke, ciclo, disculpa
inherit
  SCREENMAN;
  BASIC_ROUT;
  ARGUMENTS;
  EXCEPTIONS
  rename
    ignore as e_ignore
feature
  map: MENUS;
  key_stroke: COMMAND;

```

```

Create is
do
    -- Inicializa la terminal.
    set_device;
    if terminal_set then
        map.Create;
        map.set_up;
        ciclo;
        map.move_cursor (map.height, 1)
    end;
    reset_device -- Re-inicializa la terminal.
rescue
    reset_device;
    disculpa
end; -- Create

-- Ciclo de lectura de una opcion del menu y la ejecucion del
-- comando correspondiente.
ciclo is
require
    not map.Void
local
    last_choice, adios: COMMAND;
    quit_state: QUIT
do
    from
        quit_state.Create;
        adios := quit_state;
        map.display;
        map.move_cursor (1, 1)
    until
        not key_stroke.Void and then
key_stroke.equal (adios)
    loop
        map.prepare_choice_menu;
        -- Despliega el menu principal
        map.display;
        map.choice_menu.hide;
        -- Introduccion de la seleccion del usuario
        from
            key_stroke :=
map.choice_menu.user_answer
        until
            map.choice_menu.submenu.Void
        loop
            key_stroke :=
map.choice_menu.user_answer
        end;
        if not key_stroke.Void then
            -- Creacion y ejecucion del comando
            last_choice.Clone (key_stroke);
            last_choice.set_target (map);
            last_choice.execute;

```

```

        map.move_cursor (map.height, 1)
    end;
    map.display
end
end; -- ciclo

disculpa is
    -- Cuando sucede algo inesperado.
do
    io.putstring ("Lo siento, el ultimo comando
fallo!");
    io.new_line
end; -- disculpa

end -- class TDA

```

**ARCHIVO: TEMPORALES.E**

```

class TEMPORALES export
    execute, target, has_frame

```

```

inherit

```

```

    COMMAND;
    MAP_ELEMENT;

```

```

feature

```

```

    lin,ale,aco,ope : STRING;
    pila:PILAS;
    cola: COLAS;
    anillo: ANILLOS;

```

```

    -- Abre y despliega la ventana donde se realiza el
    cuestionario

```

```

    -- que selecciona el tda adecuado

```

```

    execute is

```

```

do
    pila.Create;
    cola.Create;
    anillo.Create;
    attach(target);
    set_height(15);
    set_width( 55);
    put_at(2,2);
    set_priority(1);
    display;
    menu;
    ciclos;
    display;
    seleccion;
end; -- execute;

```

```

-- Crea el menu del cuestionario
menu is
do
    textual.Create(500);
    textual.append("          ORDENAMIENTO TEMPORAL \n\n");
    textual.append(" - Estructura lineal      (s/n) _
\n");
    textual.append(" - Acceso aleatorio      (s/n) _
\n");
    textual.append(" - Estructura acotada    (s/n) _
\n");
    textual.append(" - Inserciones y eliminaciones \n\n");
    textual.append(" por un solo extremo    (s/n) -
\n");

    show;
    display;
    set_frame
end; -- menu

-- Preguntas del cuestionario
ciclos is
do
    move_cursor(4,36);
    echo;
    display;
    from
    lin.Create(1);
    until lin.equal("s") or lin.equal("n")
    loop
        read_string(1);
        lin.append(last_string);
        if not (lin.equal("s") or lin.equal("n") ) then
            bell;
            move_cursor(4,36);
        end; -- if
    end; --loop

    move_cursor(5,36);
    echo;
    display;
    from
    ale.Create(1);
    until ale.equal("s") or ale.equal("n")
    loop
        read_string(1);
        ale.append(last_string);
        if not (ale.equal("s") or ale.equal("n") ) then
            bell;
            move_cursor(5,36);
        end; -- if
    end; --loop

```

```
move_cursor(6,36);
echo;
display;
from
  aco.Create(1);
until aco.equal("s") or aco.equal("n")
loop
  read_string(1);
  aco.append(last_string);
  if not (aco.equal("s") or aco.equal("n") ) then
    bell;
    move_cursor(6,36);
  end; -- if
end; --loop

if lin.equal("n") then
  ope.Create(1);
  ope.append("n");
else
  move_cursor(8,36);
  echo;
  display;
  from
    ope.Create(1);
  until ope.equal("s") or ope.equal("n")
  loop
    read_string(1);
    ope.append(last_string);
    if not (ope.equal("s") or ope.equal("n") ) then
      bell;
      move_cursor(8,36);
    end; -- if
  end; --loop
end; -- if

noecho;
end; -- ciclo

-- Selecciona el tda adecuado
seleccion is
  local vent : MENUS;
         leyenda : STRING;
         com      : COMMAND;
do
  vent.Create;
  vent.set_height(4);
  vent.set_width( 50);
  vent.put_at(14, 7);
  leyenda.Create(100);
  leyenda.append("SELECCION ADECUADA : \n");
```

```

    if ope.equal("s") and lin.equal("s") then
    -- pilas
    com.clone(pila);
    if aco.equal("s") and ale.equal("s") then
    leyenda.append("\nPILAS ACOTADAS IMPLEMENTADAS CON
ARREGLOS\n");
    end; --if
    if (aco.equal("s") and ale.equal("n") )then
    leyenda.append("\nPILAS ACOTADAS IMPLEMENTADAS CON
LISTAS LIGADAS\n");
    end; --if
    if aco.equal("n") and ale.equal("s") then
    leyenda.append("\nPILAS IMPLEMENTADAS CON LISTAS
LIGADAS\n");
    else
    leyenda.append("\nPILAS IMPLEMENTADAS CON ARREGLOS\n");
    end; --if
    end; --if

    if ope.equal("n") and lin.equal("s") then
    -- colas
    com.clone(cola);
    if aco.equal("s") and ale.equal("s") then
    leyenda.append("\NCOLAS ACOTADAS IMPLEMENTADAS CON
ARREGLOS\n");
    end; --if
    if aco.equal("s") and ale.equal("n") then
    leyenda.append("\NCOLAS ACOTADAS IMPLEMENTADAS CON
LISTAS LIGADAS\n");
    end; --if
    if aco.equal("n") and ale.equal("s") then
    leyenda.append("\NCOLAS IMPLEMENTADAS CON LISTAS
LIGADAS\n");
    else
    leyenda.append("\NCOLAS IMPLEMENTADAS CON ARREGLOS\n");
    end; --if
    end; --if

    if ope.equal("n") and lin.equal("n") then
    -- anillos
    com.clone(anillo);
    if aco.equal("s") and ale.equal("s") then
    leyenda.append("\NANILLOS ACOTADOS IMPLEMENTADOS CON
ARREGLOS\n");
    end; --if
    if aco.equal("s") and ale.equal("n") then
    leyenda.append("\NANILLOS ACOTADOS IMPLEMENTADOS CON
LISTAS LIGADAS\n");
    end; --if
    if aco.equal("n") and ale.equal("s") then
    leyenda.append("\NANILLOS IMPLEMENTADOS CON LISTAS
LIGADAS\n");
    else

```



```
leyenda.append("\nANILLOS IMPLEMENTADOS CON
ARREGLOS\n");
end; --if
end; --if
vent.priority(1);
vent.associate_text(leyenda);
vent.display;
vent.set_frame;
display;
show;
io.readchar;
vent.put_at(2,2);
vent.associate_text(" ");
vent.hide;
com.set_target(vent);
com.execute;
remove_text;
hide;
unset_frame;
end; -- seleccion
end
```

## ARCHIVOS DE INFORMACION

## AXIOMAS

## ARCHIVO: ANILLO.AXI

## Axiomas de Anillos

```

IsEmpty( Empty() ) = TRUE
IsEmpty( Add( r,i ) ) = FALSE
Head( Empty() ) = error
Head( Add( q,i ) ) = si IsEmpty( q ) entonces i
                    otro Head( q )

```

donde i es un elemento y r es un anillo.

## ARCHIVO: ARBOL.AXI

## Axiomas de Arboles

```

IsEmptyTree( EmptyTree() ) = TRUE
IsEmptyTree( ConstTree( i,l,r ) ) = FALSE
Root( EmptyTree() ) = error
Root( ConstTree( i,l,r ) ) = i
LeftTree( EmptyTree() ) = error
LeftTree( ConstTree( i,l,r ) ) = l
RightTree( EmptyTree() ) = error
RightTree( ConstTree( i,l,r ) ) = r

```

donde i es un elemento, y r y l son arboles.

## ARCHIVO: COLA.AXI

## Axiomas de Colas

```

IsEmpty( Empty() ) = TRUE
IsEmpty( Add( q,i ) ) = FALSE
Front( Empty() ) = error
Front( Add( q,i ) ) = si IsEmpty( q ) entonces i
                    otro Front( q )
Back( Empty() ) = error
Back( Add( q,i ) ) = si IsEmpty( q ) entonces Empty()
                    otro Add(Back(q),i)

```

donde i es un elemento y q es una cola.

**ARCHIVO: CONJUNTO.AXI**

## Axiomas de Conjuntos

```

Insert( i,s )= si Contain( i,s ) entonces s
                otro          Insert( i,s )
IsEmpty( Empty() ) = TRUE
IsEmpty( Insert( i,s ) ) = FALSE
Contain( i,Empty() ) = FALSE
Contain( j,Insert( i,s ))= si i=j entonces TRUE
                otro          Contain( j,s )
Remove( i,Empty() ) = Empty()
Remove( j,Insert( i,s ))= si i=j entonces Remove( j,s )
                otro          Insert( i,Remove( j,s ))
Union( s,Empty() ) = s
Union( s,Insert( i,t ))= Insert( i,Union( s,t ) )
Intersection( s,Empty() ) = Empty()
Intersection( s,Insert(i,t) )= si Contain( i,s ) entonces
                Insert( i,Intersection( s,t ) ) otro Intersection( s,t )
Difference( s,Empty() ) = s
Difference( s,Insert(i,t) )= si Contain( i,s ) entonces
                Remove( i,Difference( s,t ) ) otro Difference( s,t )
Size( Empty() ) = 0
Size( Insert( i,s ) ) = 1 + Size( Remove( i,s ) )

```

donde i y j son elementos y s y t son conjunto.

**ARCHIVO: LISTA.AXI**

## Axiomas de Listas

```

IsEmpty( Empty() ) = TRUE
IsEmpty( Cons(h,t) ) = FALSE
Head( Cons(h,t) ) = h
Tail( Cons(h,t) ) = t
Head( Empty() ) = ERROR
Tail( Empty() ) = ERROR

```

donde h es un elemento y t es una lista.

**ARCHIVO: COLA.AXI**

Axiomas de Pilas

```
IsEmpty( Empty() )= TRUE
IsEmpty( Push( s,i ) )= FALSE
Top( Empty() )= error
Top( Push( s,i ) )= i
Pop( Empty() )= error
Pop( Push( s,i ) )= s
```

donde i es un elemento y s es una pila.

**ARCHIVO: TABLA.AXI**

Axiomas de Tablas

```
IsEmpty(Empty())= TRUE
IsEmpty(Insert(T,k,i))= FALSE
Retrieve(Insert(T,k,i),j)= si j=k entonces i
                           otro Retrieve(T,j)
Retrieve(Empty(),i)= error
IsIn(Empty(),j)= FALSE
IsIn(Insert(T,k,i),j)= si j=k entonces TRUE
                       otro IsIn(T,j)
```

donde T es una tabla, k y j son numeros cardinales e i es un dato asociado a k.

**CONCEPTOS**

**ARCHIVO: ANILLO.CON**

Conceptos de Anillos

Una cola es una secuencia de cero o mas elementos arreglados de una manera circular.

Los elementos pueden ser adicionados o removidos por un solo punto llamado cabeza o primer elemento del anillo.

Un anillo puede ser graficamente representado como un conjunto de elementos, cada uno de los cuales tiene un sucesor y un predecesor. La extencion de un anillo representa el numero de elementos que existe en el. Si no existe ningun elemento se dice que el anillo esta vacio.

**ARCHIVO: ARBOL.CON**

Conceptos de Arboles

Un arbol es una coleccion de nodos que pueden tener un numero arbitrario de referencias hacia otros nodos, donde no pueden existir referencias de tipo ciclo o corto circuito; para cada dos nodos existe un unico camino que los conecta. Si un arbol dado no tiene nodos, se lo considera nulo. Asociado a cada nodo esta un elemento, este es el valor del nodo.

Existe un nodo que se dice que es la raiz del arbol. A un arbol con raiz se le llama arbol orientado. La coleccion de arboles se conoce como bosque.

Si un nodo hace referencia a otros, se dice que es el padre de los nodos subordinados; un nodo subordinado es el hijo del padre.

Si un nodo dado no tiene hijos, se le llama hoja o terminal. los hijos de un nodo dado son asimismo raices de arboles subordinados o subarboles de su padre.

El nivel de un nodo dado es la medida de la longitud de este a lo largo del unico camino hacia la raiz, el cual por definicion tiene nivel 0.

**ARCHIVO: COLA.CON**

Conceptos de Colas

Una cola es una secuencia de elementos en la cual los elementos son adicionados por un extremo (llamado fin de la cola) y son removidos por el otro extremo (llamado cabeza).

La cola puede ser representada graficamente como una lista lineal que crece por un extremo y se extingue por el otro.

La longitud de la cola representa el numero de elementos que esta contiene. Si no existe ningun elemento, se dice que la cola esta vacia.

**ARCHIVO: CONJUNTO.CON**

Conceptos de Conjuntos

Un conjunto es una coleccion de elementos extraidos de una clase de objetos llamado Universo. Esta coleccion no puede contener ningun elemento duplicado. La extencion de un conjunto representa el numero de elementos que hay en el conjunto. Si no existen elementos en el conjunto, se lo considera vacio.

Si un elemento pertenece a un conjunto, se considera que el elemento es un miembro del conjunto. El conjunto A es llamado un subconjunto del conjunto B, si y solo si todos los elementos de A son tambien miembros de B, en tal caso tambien se puede decir que B es un superconjunto de A. Si adicionalmente A es diferente de B, entonces A es un subconjunto propio de B (y B un superconjunto propio de A).

**ARCHIVO: LISTA.CON**

Conceptos de Listas

Una lista es una secuencia de cero o mas elementos en la cual los elementos pueden ser adicionados o removidos de o en cualquier posicion de manera que un orden lineal es mantenido.

**ARCHIVO: PILA.CON**

Conceptos de Pilas

Una pila es una secuencia de elementos que son adicionados y removidos por el mismo extremo llamado tope de la pila.

Una pila se puede representar graficamente como una lista lineal que crece y se extingue por el mismo extremo.

La profundidad de la pila representa el numero de elementos en ella. Si no hay elementos en la pila, se dice que esta vacia.

**ARCHIVO: TABLA.CON**

Conceptos de Tablas

Una tabla es una coleccion de un numero arbitrario de distintos elementos (todos del mismo tipo), cada uno de los cuales es identificado por una llave distinta.

Los elementos en una tabla consisten de muchas partes diferentes, una de las cuales es elegida como llave, es decir, es utilizada para identificar un elemento.